AFIT/GST/ENS/93M-7

PATIENT SCHEDULING AND AIRCRAFT
ROUTING FOR STRATEGIC
AEROMEDICAL EVACUATION

THESIS

Michael J. Loftus
Major, USAF

AFIT/GST/ENS/93M-7

Approved for public release; distribution unlimited

AFIT/GST/ENS/93M-7

PATIENT SCHEDULING AND AIRCRAFT ROUTING FOR STRATEGIC AEROMEDICAL EVACUATION

THESIS

Presented to the Faculty of the School of Engineering

of the Air Force Institute of Technology

Air University

In Partial Fulfillment of the

Requirements for the Degree of

Master of Science in Operations Research

Michael J. Loftus, B.S.

Major, USAF

March 1993

Approved for public release; distribution unlimited

THESIS APPROVAL

STUDENT: Michael J. Loftus **CLASS:** GST-93M-7

THESIS TITLE: Patient Scheduling and Aircraft Routing for Strategic Aeromedical Evacuation

DEFENSE DATE: 26 Feb 93

COMMITTEE	NAME/DEPARTMENT	SIGNATURE
Advisor	Maj John J. Borsi/ENS	
Reader	LtC James T. Moore/ENS	James T. Moore

Preface

I would like to take this opportunity to express my appreciation to all of those who have supported and helped me through this effort. The personnel at Air Mobility Command, most notably LtC Joe Litko, Keith Ware, and Maj. Phil Mahlem were extremely helpful providing the initial idea and data required to complete the research effort.

I would also like to express my sincere appreciation to my advisor Maj. John Borsi for his invaluable guidance throughout my time at AFIT. I would also like to thank LtC Jim Moore, my reader/co-advisor, for all his help and comments in the preparation of the final document. I greatly appreciate both of their efforts.

Most of all, I would like to thank my family for putting up with the late hours and lost weekend time. If it were not for my wife's continual support and encouragement, I would not have been able to finish this project. Thank you Glee. And finally, I would like to thank my daughters, Michelle and Amy, for putting up with me. Maybe now I won't be such a "grump" all the time.

Michael J. Loftus

Table of Contents

List of Tables

List of Figures

Abstract

The objective of this study is to develop an approach that would assign patients to aeromedical evacuation aircraft and route each aircraft to a single CONUS airport. The goal is to minimize total patient wait time.

A heuristic algorithm is developed which exploits the structure of the problem. The first subproblem solved is the assignment of aircraft to airports. This subproblem is solved using integer programming and the results are input into the second subproblem. The second subproblem is the patient to aircraft assignment problem. This subproblem minimizes patient wait time and is solved using network flow programming. The two subproblems are linked into one algorithm and solved iteratively until termination criteria are met.

The algorithm is tested on three data sets. The results indicate that the algorithm is an efficient method for scheduling patients and routing aircraft, although optimality is not guaranteed.

PATIENT SCHEDULING AND AIRCRAFT ROUTING FOR STRATEGIC AEROMEDICAL EVACUATION

1. *INTRODUCTION*

A critical task during periods of war is the aeromedical evacuation of injured soldiers from the war zone. The morale of the troops, and the morale of the nation, is influenced by the ability to get the wounded back to the United States quickly.

The policy of the Department of Defense is to move the casualties by aircraft. The responsibility for the aeromedical evacuation mission falls to the U.S. Air Force's Air Mobility Command (AMC).

1.1 Background

Air Mobility Command is responsible for providing a worldwide aeromedical evacuation system for the United States Armed Forces (22:3). The system provides for airlift within the continental U.S. (CONUS), between theaters of operation, and within each theater. Within the theater of operation, medical resources are distributed among the services according to functional levels called echelons.

Higher echelons are identified by an increasing level of medical care for the wounded soldier. The first echelon (1E) is located along the line of contact with the enemy where only limited care is available. The wounded soldier is moved by foot, ground, or air transportation to the second echelon (2E). The second echelon is a holding area responsible for emergency/urgent casualty care. If soldiers cannot be returned to duty within a specified period of time, they move to the third echelon (3E). The time period and other specifics of the 1E and 2E vary because they are the responsibility of the soldier's parent service (Army, Air Force, Navy/Marines). The parent services are also responsible for patient movement from 2E to 3E; however, the Air Force may move patients between these levels if aircraft are available and conditions permit.

The patient becomes the responsibility of the Aeromedical Evacuation System at the third echelon. The 3E is further back from the combat zone than 1E and 2E and provides more sophisticated medical care. If patients cannot be returned to duty within a specified period of time (a longer time period than when at 1E and 2E), they will be evacuated to the fourth echelon (4E). The 4E is a level that is removed from the combat zone in an area called the communications zone (CCMMZ). The COMMZ contains "lines of communication, establishments for supply and evacuation, and

other agencies required for the immediate support and maintenance of all field agencies" (22:4). COMMZ also provides the connecting link between the combat zone and the CONUS. Transportation between 3E and 4E is the responsibility of the Air Force and is generally performed by C-130 aircraft, although C-9 and C-141 aircraft may also be used. Each aircraft has dedicated medical equipment and medical personnel on board. From the 4E level, the patient is either returned to duty or evacuated to the United States. Under current plans, movement to the CONUS will be performed by C-141 aircraft. Once in the CONUS, the patient is moved by C-9 or ground transportation to a hospital which is the patient's final destination.

The movement of patients between echelons and from 4E to the CONUS is controlled by medical personnel called regulators. Regulating is defined as "the selection of a source of care to which casualties are evacuated" (22:5). Medical regulators match casualties with hospitals capable of providing required medical care. Regulation within the combat zone (1E, 2E, 3E) is the responsibility of each individual service. The regulation of casualties from the combat zone to the COMMZ (3E to 4E) and from the COMMZ to the CONUS is accomplished by the Armed Forces Medical Regulating Office. One of the functions of a medical regulator

at the 4E level is to categorize patients based on type of injury. The eight categories are:

1) General Medical;
2) Psychological;
3) Surgical Medical;
4) Orthopedic;
5) Spinal Injury;
6) Burn Injury;
7) Pediatrics;
8) OB/GYN.

The regulator assigns a patient to a specific hospital bed in the United States. The assignment is based on the medical care required and hospital bed availability for that patient's injury category. For example, a burn patient would be assigned to a hospital bed in the U.S. only if that hospital can provide appropriate care for a burn patient. The actual hospitals and number of available beds have been identified by the National Disaster Medical System (NDMS). The NDMS was developed to respond to any national disaster to include a miliary contingency. As of July 1986, 73,000 beds in 950 hospitals have been identified (5:39). Depending on availability, patients are assigned to hospitals in this order: military hospitals, Veterans Administration hospitals and civilian hospitals.

1.2 Problems With the Present System

Due to the lack of a dedicated intratheater aircraft in the present system, the United States now has a sizable strategic aeromedical evacuation shortfall. The U.S. depends on C-141s for the movement of patients from the COMMZ to CONUS, but the C-141 is also the primary aircraft used for cargo airlift by the Armed Forces. In 1984, the shortfall of C-141s was identified when the Military Airlift Command (predecessor of AMC) commissioned a Patient Distribution-Redistribution Study (PDS). The PDS compared the need for aeromedical airlift with the planned cargo flow in a European scenario. The results showed an insufficient number of C-141 airframes for the aeromedical evacuation mission (10:8). Based on this study, the Office of the Secretary of the Air Force authorized the creation of a new segment of the Civil Reserve Air Fleet (CRAF) dedicated to aeromedical evacuation.

1.3 Future System

The future aeromedical evacuation system is based heavily on CRAF aircraft. Boeing 767s will replace the C-141 for patient movement from the COMMZ to the CONUS. This will allow the C-141 to be dedicated to its primary

mission of cargo airlift. The Boeing 767s will transport patients from the COMMZ to major hubs within the CONUS. Within the CONUS, C-130s will redistribute the patients from the major hubs to smaller airports closer to the patient's assigned hospital bed. The C-130s will replace the C-9s so that the C-9s may deploy to the combat zone and augment theater evacuation forces. If the C-9s do not deploy, they will augment the C-130s, performing patient redistribution in the CONUS (5:31).

AMC is studying options to improve the efficiency of the aeromedical evacuation system. One such option is to bypass the major hubs and to deliver patients directly to the airports nearest their assigned hospital. This option would decrease the stateside redistribution of patients. This approach would save time as the patients would get to their assigned hospital more quickly than under the present system. It would also eliminate much of the required handling of the patients. Presently, a patient arriving in the CONUS has to be unloaded at a major hub, reloaded on a smaller aircraft, flown to another airfield, and unloaded a final time. Flying direct would eliminate two steps in this process and save time and resources.

The elimination of the major hubs is not a goal of this research. Instead, this research concentrates on the medical regulator's task of assigning patients to an ultimate

destination and determining the routing of evacuation aircraft. As such, these results are applicable to both the current hub and spoke system and the proposed direct delivery system.

1.4 Problem Statement

Air Mobility Command does not have a method to schedule patients on individual aircraft which would fly direct from the 4E COMMZ to U.S. airports. The purpose of this research is to develop a method to schedule patients on Boeing 767s and then route each aircraft to a single CONUS airport so that the total time all patients spend waiting for evacuation is minimized. The enroute time to be minimized for each patient starts when medical personnel release that patient for evacuation from the COMMZ and stops when that patient is loaded on a Boeing 767 for transport to the CONUS.

1.5 Scope and Assumptions

There may be multiple strategic evacuation points in the COMMZ; however, in this research we will consider the regulation problem at a single COMMZ evacuation point. The problem addressed is the problem that each medical regulator faces: who to put on what aircraft and where to send that

aircraft. The job of the regulator is vital to the smooth flow of patients in the system. The ability to schedule patients on aircraft and the routing of those aircraft in an efficient manner is crucial if the concept of direct delivery is to be effectively implemented.

There are two versions of the Boeing 767 that are being purchased for the CRAF: the 200ER and the 300ER. Each version has the capability to carry 111 litter patients. However, patients of certain categories require medical equipment which utilizes more than one litter position. Examples of this type of equipment are ventilators, cardiac monitors and stryker frames. Headquarters AMC estimates that there will be a total of 11 litter positions taken up by this medical equipment on each flight (15:2). Therefore, each Boeing 767, regardless of type, is assumed to have a capacity of 100 litter patients and the need for this additional equipment will not be explicitly modeled. The major difference between the 200ER and 300ER is the capacity to carry ambulatory patients. Ambulatory patients do not require a litter for the flight and will sit in a different portion of the aircraft. The 200ER can only carry two ambulatory patients while the 300ER can carry up to 75 ambulatory patients. AMC is planning on procuring a fleet of 44 Boeing 767s having a mix of 60 percent 200ER and 40 percent 300ER (15:1). Therefore, the assumed fleet size of

Boeing 767s used in this research is 26 200ERs and 18 300ERs.

The range of each type aircraft is assumed to be 4400 nautical miles (15:2). The average cruise speed is also assumed to be identical: 460 nautical miles/hour. Based on these two assumptions, the time required to fly to any airport from the COMMZ evacuation point is assumed not to be a factor. While the time of flight frcm the combat zone to the various airports will be different, the differences in travel time are assumed not to be large enough to be a factor and will be ignored. The time frame minimized ends when the patient is loaded on the aircraft. In this research, the aircraft are assumed to fly from the COMMZ evacuation point directly to an airport in the U.S. without stopping enroute. This assumption holds except for a Far Eastern scenario where a refueling stop in Hawaii is necessary. Even with a refueling stop, the overall time required is not affected since each aircraft would have to stop.

Another assumption is that only Boeing 767s will be available for the aeromedical evacuation mission. While there are scenarios in which both C-141s and Boeing 767s would be used, C-141s are assumed not to be available for the aeromedical evacuation mission. This is consistent with AMC's present plan of evacuation.

The percentage of patients in each injury category is assumed to match the percentages shown in historical data. Data has been obtained from AMC which details the expected percentages and is shown in Table 1. Also included in the table is the average time a patient spends recovering in U.S. hospitals.

Table 1

Percentage of Injury by Category (15:4)

INJURY CATEGORY	PERCENTAGE	AVG RECOVERY TIME
General Medical	12.6	16
Psychological	3.2	29
Surgical	44.1	24
Orthopedic	36.8	50
Burn	2.6	33
Spinal	0.7	38
Pediatric	0.0	N/A
OB/GYN	0.0	N/A

The last two categories, pediatric and OB/GYN, will be disregarded since it is generally assumed that there will be very few patients in these categories.

The time a patient is loaded on an aircraft is assumed to be the same as the arrival time of the aircraft in the theater of conflict. The loading of the aircraft will take time, but the time will be the same for every patient as the aircraft cannot take off until all patients are loaded and

is assumed to be approximately the same for all plane loads. Therefore, the loading time does not effect the solution of the problem.

Another major assumption is that the medical regulators know the arrival times of the Boeing 767s in advance. The medical regulators are assumed to know the number and location of available hospital beds by category in the CONUS. The beds have been identified by the National Disaster Medical System plan discussed earlier. Medical regulators need this information so they can make decisions on who to put on each aircraft and where to send each aircraft.

The final assumption made is that there is a fixed number of CONUS airports, identified by military and civilian planners.

1.6 Overview

The remaining chapters detail the research effort. Chapter 2 contains an overview of the mathematical programming literature that relates to this problem. In Chapter 3, the aeromedical evacuation problem is formulated as a mixed integer program (MIP). A small problem is formulated and solved and a discussion of the applicability of the model to a larger problem is discussed. In Chapter 4, a heuristic algorithm is presented and the development and performance

of the algorithm is described. Finally, conclusions and recommendations are presented in Chapter 5.

2. LITERATURE REVIEW

The purpose of this chapter is to discuss the information found in the literature which applies to this research effort. In this chapter, information pertaining to aeromedical evacuation, computational complexity, scheduling theory, integer programming, network theory, and goal programming will be reviewed.

2.1 Aeromedical Evacuation

Most of the literature specifically addressing aeromedical evacuation deals with the policies of aeromedical evacuation and the number of aircraft required to perform the mission.

Four masters theses have been conducted at the Air Force Institute of Technology concerning the war time aeromedical evacuation of patients. Joseph Alfano and John O'Neill performed a simulation which uses a European scenario to test the capability of CRAF aircraft. However, their main focus was on the route structure for the CONUS redistribution of patients. Their simulation used a hub and spoke system, and they concluded that there was an insufficient number of C-9 aircraft. They mentioned the direct delivery of patients only as an alternative to CRAF aircraft routing (1:31). Michael Burns and W. Brand Carter conducted

parallel research efforts in 1990. Their main focus was also on the CONUS redistribution of patients. Charles Wolfe has conducted a simulation study to evaluate the entire aeromedical evacuation system using Boeing 767 aircraft and a hub and spoke system. It is titled "The Use of Simulation to Evaluate Strategic Aeromedical Evacuation Policy and Planning"(21).

Headquarters AMC has discussed alternatives to the present hub and spoke system, but has not conducted an in-depth study on the matter. These alternatives were mentioned in a briefing to the AMC staff on alternatives to the MD-80 aircraft (19:19).

Lt Col John D. Becker's Air War College Research paper, "Aeromedical Evacuation: Do the Pieces Fit", provides an excellent overview of the entire aeromedical evacuation process. Becker first describes the theater aeromedical evacuation system and what impact changing the time period that the patient spends in the COMMZ has upon aeromedical planning and policy. The theater evacuation policy determines this time period which influences the percentage of patients that have to be evacuated to the CONUS.

Becker reiterates the C-141 shortfall identified in the Patient Distribution-Redistribution Study commissioned by MAC. He then discusses the proposed concept of a hub and spoke operation using the Boeing 767. Lt Col Becker does

not discuss the concept of alternatives to the hub and spoke system.

2.2 Complexity Theory

This research involves the routing of aircraft and the scheduling of patients. According to Bodin, most routing and scheduling problems fall into a category of problems known as non-deterministic polynomial hard (NP-Hard) (6:76). A review of the concepts of complexity theory is required to understand approaches for solving problems in the class NP-Hard.

The goals of complexity theory are to broadly classify problems and algorithms according to the time needed to solve them on computers (14:3). Decision problems come in all degrees of difficulty, from ones that are solved readily to ones that cannot be solved. The best solved problems are classified Polynomial (P). The class P is the set of problems for which the number of basic computational steps required to generate a solution is a polynomial function of the size of the problem (14:8). Some of the problems that fall in the class P are linear programming, minimum spanning trees, and maximum flow in a network (14:8).

The class P is a subset of problems of the class NP. A problem is in NP if one can verify the correctness of a

solution in polynomial time. The class NP contains all class P problems.

Problems to which all members of NP polynomially reduce are called NP-Hard. A problem reduces to a second problem if, for every instance of the first problem, we can construct an equivalent instance of the second problem. Therefore, any algorithm that solves the second problem can be converted to an algorithm for solving the first problem. A problem polynomially reduces to the second problem if a polynomial time algorithm for the second would imply a polynomial time algorithm for the first (14:6). Problems which are both in NP and NP-Hard are called NP-Complete. Problems of this class are among the most difficult to solve. If a problem is NP-Hard, it is as least as difficult as any NP-Complete problem (13:138).

Generally, the effort required to solve NP-Hard problems increases explosively with problem size in the worst case for all known solution algorithms (6:76). If a problem has been proven to be NP-Hard, it does not preclude the existence of polynomial time algorithms for specific cases of the problem. When faced with an NP-Hard problem that cannot be solved with a polynomial algorithm, a heuristic procedure is often employed in an attempt to find a feasible solution. Nemhauser and Woolsey define a heuristic algorithm as an approximate algorithm designed to find good, but

not necessarily optimal, solutions quickly (13:393). A heuristic is considered effective if the solutions are consistently close to optimal. A heuristic algorithm usually exploits some aspect of the problem structure (either mathematically or intuitively) to provide feasible and near optimal solutions (6:76).

2.3 Scheduling Theory

This research effort deals with scheduling patients on individual aircraft to minimize their wait time. Many aspects of scheduling theory, particularly priority dispatching rules apply directly to this aspect of the problem. Kenneth Baker in his book, *Sequencing and Scheduling*, lists and explains some of the most popular priority dispatching rules. They are (2:197,217):

1) Shortest Processing Time (SPT): Select the operation with the minimum processing time.

2) First Come First Serve (FCFS): Select the operation that entered earliest.

3) Random (RANDOM): Select the operation at random.

4) Most Work Remaining (MWKR): Select the operation associated with the job having the most work remaining to be processed.

5) Least Work Remaining (LWKR): Select the operation associated with the job having the least work remaining to be processed.

6) Earliest Due Date (EDD): Select the operation with the earliest due date.

No single dispatching rule has been found to be dominant in all cases. The priority dispatching rule used depends on the type problem and the measure of effectiveness being optimized.

Analyzing the aeromedical evacuation problem reveals that some of the priority dispatching rules need not be considered. In Chapter 1, the assumption was made that the work remaining (flying time of the aircraft) was identical for all patients; therefore, MWKR and LWKR can be eliminated from the list of dispatching rules to be considered. The patients do not have a due date at a CONUS hospital, so EDD may also be eliminated from the list. The three remaining dispatching rules (SPT, FCFS, and RANDOM) are all possible candidates for use as a priority dispatching rule for the aeromedical evacuation problem.

2.4 Integer Programming

Both routing and scheduling problems may also be formulated as integer programs in many, but not all, cases. Integer programming (IP) techniques are often applied to solve routing and scheduling problems. The purpose of this section is to discuss some of the solution techniques used to solve integer programs, including branch and bound, and the Pivot and Complement heuristic.

Perhaps the method to solve integer programs that is most widely used is the branch and bound method. The branch and bound method finds the optimal solution by efficiently enumerating the points in a problem's feasible region (20:488). The branch and bound method begins by solving the linear programming (LP) relaxation to the IP. The value of the objective function of the LP relaxation provides an upper bound when the goal is to maximize the objective function. No integer solution can exceed the upper bound. If the solution to the LP relaxation contains all integer values, then the solution to the LP is also the solution to the IP (20:489). If an integer solution is not found, the next step is to partition the feasible region in an attempt to find the optimal solution of the IP. This is the branching portion of the procedure. An integer variable that is not integer valued in the current solution of the relaxed LP is picked to branch upon. Two or more subproblems are created and constraints are then added to the problem to eliminate the non-integer solution. The process is repeated until all integer variables have integer values. The objective function value of this solution is a lower bound on the optimal solution value for the problem. The next step is to enumerate other possible combinations in an attempt to find an improved integer solution. There are different techniques which pick the branching variable and how to search

each branch. Some of the techniques are depth first, breadth first, and quick improvement. Normally, depth first is used because feasible solutions satisfying the integer requirements are generally found deeper in the branch and bound tree and it is computationally easier when using the dual simplex procedure (13:358).

Binary integer programs and mixed integer programs are in the class NP-Hard (14:83). Therefore, as the problems get larger, the computational time required to solve them increases greatly. If the program gets too large, other methods such as heuristic procedures should be used in order to solve the problem. One heuristic is the Pivot and Complement heuristic developed by Egon Balas and Clarence Martin.

The Pivot and Complement heuristic is used for finding approximate solutions to 0-1 programming problems. It uses the fact that a 0-1 program is equivalent to the associated linear program with the added requirement that all slack variables, other than those in the upper bounding constraints, be basic (3:86). The method starts by solving the LP relaxation, and then searches in the vicinity of the optimal solution to the LP relaxation for an integer feasible solution. It conducts this search by trying to force any basic integer variable that is fractional out of the basis. This is done by performing pivots. Non-basic inte-

ger variables can be flipped from zero to one or from one to zero. This is called complementing (7:227). Both one at a time and two at a time complements are attempted. If a feasible solution is found, additional complements are performed in an attempt to improve it. The computational effort involved in the Pivot and Complement heuristic procedure is bounded by a polynomial in the number of constraints and variables (3:89).

2.5 Network Theory

Many routing and scheduling problems can be formulated as networks. The advantage to formulating problems as networks is that many network problems can be solved in polynomial time (6:75).

A network or graph is defined using two types of sets: nodes and arcs. An arc consists of an ordered pair of vertices (nodes) and represent a possible direction of motion that may occur between vertices (20:389). Networks can be represented pictorially by points and lines. The points represent the nodes and the lines represent the arcs.

Minimum cost flow programming is the type of network flow programming that is used in this research. In a minimum cost flow program, the arcs of the network are described by two parameters. The parameters are the cost of the arc and the capacity of the arc. The cost of the arc reflects

the expense for a unit of flow to travel along that arc. The capacity of the arc is how much flow can travel through the arc. The objective of minimum cost flow programming is to flow the available supply through the network to satisfy demand at minimal cost (4:420).

One of the attractive aspects of formulating the problem as a network is the structure of the constraint matrix. The constraint matrix is called a node-arc incidence matrix. When formulated this way, the nodes of the problem are represented by rows of the matrix. The columns of the matrix represent the arcs of the problem. A matrix is totally unimodular if the determinant of each square submatrix is 0, 1, or -1 (13:540). If a matrix is totally unimodular and the model parameters are integral, then the solution to the LP relaxation will result in integer answers (13:541). This aspect is advantageous because one does not need to use an IP solution package to get integer answers.

The network simplex algorithm is a specialization of the simplex algorithm. It performs simplex operations directly on the network itself. The overall efficiency with this procedure is 200-300 times faster than the standard simplex procedure. This means that large problems can be solved with a reasonable amount of effort (4:419).

If a network structure is embedded in a linear program, the problem is called a network with side constraints. In

general these problems do not have a totally unimodular matrix, and cannot be directly solved by LP procedures. When the network part of the problem is large compared to the non-network part, especially if the number of side constraints is small, it is worthwhile to exploit this structure in the solution process. The solution time will be reduced when solved this way. However, there is no guarantee that the problem will be solved in polynomial time.

2.6 Goal Programming and Deviational Variables

Goal programming is a technique used to formulate and solve linear programs where there are multiple objectives or conflicting goals. Several aspects of goal programming are used in this research, particularly the use of deviational variables.

Generally, in goal programming, two deviational variables are introduced for each inequality constraint (representing a goal) in a linear program. There is a positive variable representing the amount by which a goal is exceeded, and a positive variable representing the amount by which a goal is underachieved. The inequality constraints are transformed into equality constraints by the addition of the two deviational variables. In a sense, we are adding a slack variable and an excess variable instead of just one or the other. After the deviational variables are added to the

constraints, a weight is given to the deviational variables in the objective function. The weights given to the deviational variables depend on the objective (or objectives) being optimized and vary depending on the specifics of the problem. Goal programming can be used in two different ways to represent tradeoffs between different goals. According to Winston, these are (20:189):

(1) Assign a penalty per unit deviation from each goal and use linear programming to minimize the total penalty incurred because of an unmet goal.

(2) Rank goals in priority from highest to lowest goal and use pre-emptive goal programming and goal programming simplex.

Pre-emptive goal programming and goal programming simplex are techniques specifically designed to solve goal programming problems and are not used in this research. Deviational variables are used in this research as described in (1) above.

3. PROBLEM FORMULATION

The purpose of this chapter is to present the formulation of the aeromedical evacuation problem as a mixed integer program (MIP). Following the model formulation, a small problem is presented and its solution discussed. Finally, there will be a discussion of the number of integer variables required to formulate a realistic problem and the need for a heuristic procedure for assigning aircraft to airports and assigning patients to aircraft.

3.1 Aeromedical Evacuation Model Formulation

In this problem, patients are assigned to aircraft and aircraft are assigned to airports. In the MIP formulation, all possible patient and aircraft assignments are represented by binary (0-1) variables. Assignments are constrained by both aircraft and airport capacities. Aircraft capacity as defined in Chapter 1 is the capacity of a Boeing 767. Airport capacity is defined to be the number of available hospital beds by injury category located at hospitals near that airport. The hospital capacity data used for this study is included in Appendix A. The overall objective of the model is to minimize the waiting time of the patients without exceeding either capacity.

Each patient and each aircraft has certain characteristics associated with them. The patients are classified based on their type of injury. Each patient also has a release time before which he cannot be transported by an aircraft to the CONUS. Each aircraft has a capacity and an arrival time when it will arrive in theater at the COMMZ evacuation field.

Each possible patient to aircraft assignment and each aircraft to airport assignment has an associated binary variable. If a variable representing a patient assignment to a particular aircraft is set to 1, the patient has been assigned to that particular aircraft. If not, the variable is 0 and the patient is not assigned to that aircraft. The same approach applies to aircraft to airport assignments.

Each of the variables have indices and sets associated with them. Index i identifies a particular patient in Set P_k. The set P_k contains patients grouped by injury category k. The set of all patients is P so $P = {}_{k} P_k$. Index j identifies a particular aircraft in Set J. The set J is composed of all aircraft regardless of model. The set J_i is a subset of J. Set J_i contains all aircraft whose arrival time is after patient i's release time. Index k identifies an injury category in Set K. The set K is composed of all injury categories. Index h identifies a particular airport

in Set H. The set H is composed of all airports. The variables used in the formulation are as follows:

$$x_{ij} = \begin{cases} 1 \text{ if patient } i \text{ is assigned to aircraft } j \\ 0 \text{ otherwise} \end{cases}$$

$$y_{jh} = \begin{cases} 1 \text{ if aircraft } j \text{ is assigned to airport } h \\ 0 \text{ otherwise} \end{cases}$$

z_{ijh}: Continuous variable with an upper bound of 1. Takes on a value of 1 when patient i is assigned to aircraft j which is assigned to airport h.

There is also a variable x_{iD} which represents a patient assignment to the dummy aircraft, which is described below.

In this formulation, the planning horizon is two days. A planning horizon is defined to be how far into the future one uses information in making the current decisions (17:540). In setting th planning horizon at two days, the model accounts for patients who cannot be airlifted within the planning horizon by assigning them to the dummy aircraft. The model does not take into account any effect patients not airlifted have upon the next day's process.

The following are parameters in the formulation:

Rel_i = Release time of patient i. This is the time medical personnel state that the patient is stable enough for evacuation to the United States.

Arr_j = Arrival time of aircraft j at the COMMZ evacuation point.

CA_j = Capacity of aircraft j

CH_{hk} = Hospital bed capacity of airport h for category k patients.

Model Formulation

The formulation is as follows:

$$Min \sum_{i\in P} \sum_{j\in J_i} ((Arr_j - Rel_i)x_{ij} + M*x_{iD}) \qquad (1)$$

SUBJECT TO:

$$\sum_{j\in J_i} x_{ij} + x_{iD} = 1 \qquad \forall\, i\in P \qquad (2)$$

$$\sum_{h\in H} y_{jh} = 1 \qquad \forall\, j\in J_i \qquad (3)$$

$$\sum_{i\in P} x_{ij} \le CA_j \qquad \forall\, j\in J_i \qquad (4)$$

$$\sum_{i\in P_k} \sum_{j\in J_i} z_{ijh} \le CH_{hk} \qquad \forall\, h\in H, \forall\, k\in K \qquad (5)$$

$$x_{ij} + y_{jh} - z_{ijh} \le 1 \qquad \forall\, i\in P,\ \forall\, j\in J_i,\ \forall\, h\in H \qquad (6)$$

$x_{ij} \in \{0,1\}$, $y_{jh} \in \{0,1\}$, $0 \le z_{ijh} \le 1$

for all $i \in P$, for all $j \in J_i$, for all $h \in H$

The objective function (Equation (1)) minimizes the patients' wait time and penalizes assignments made to the dummy aircraft. Wait time is defined to be the amount of time between the time the patient is released for evacuation by medical personnel and the time he is loaded on an aircraft. The actual wait time minimized by the model is the time difference between the arrival of the patient's assigned aircraft and the release time of the patient. The aircraft's arrival time and the time the patient is loaded onto that aircraft are assumed to be identical. Aside from the wait time, there is an additional term ($M*x_{iD}$) in the objective function. This term penalizes any patient assignment to the dummy aircraft. The penalty term (M) used is greater than the planning horizon length and must be greater than the greatest possible wait time of any patient.

The first set of constraints (Equation (2)) state that a patient must be assigned to exactly one aircraft.

The second set of constraints (Equation (3)) state that each aircraft must be assigned to one airport. It is assumed here that an aircraft will only offload patients at one airport. This assumption is preferred by AMC (12).

Equation (4) ensures that the number of patients assigned to each aircraft does not exceed the aircraft capacity. The dummy aircraft has an infinite capacity.

The final two sets of constraints (Equations (5) and (6)) ensure that the number of patients in an injury category transported by all aircraft going to a common airport does not exceed the available number of beds for that injury category at that location. The constraints of Equation (6) ensure that z_{ijh} is one if patient i is assigned to aircraft j that goes to airport h. These constraints were a modification of the formulation outlined by Ravindran, Phillips, and Solberg in their book *Operations Research Principles and Practice* on page 190. The z_{ijh} variable is in effect the product of the x_{ij} and y_{jh} variables (16:190).

3.2 Sample Problem Formulation and Solution.

The purpose of developing this sample problem was to formulate and solve a small, but realistic problem. Analyzing a small problem's formulation and solution helps to determine whether the solution methodology can be applied to larger, more realistic problems. The sample aeromedical evacuation problem was solved using LINDO software on a personal computer, and its formulation and solution is listed in Appendix B.

This sample problem addresses the scheduling of ten patients on two aircraft traveling to two different airports. The patients have two possible injury categories and different release times (given in days). There are two

aircraft with different capacities and different arrival times. The hospitals have different capacities for each of the two categories. The following tables show the specifics of the problem:

Table 2

Sample Problem Aircraft

Aircraft	Arrival Time	Capacity
E	4.0	5
F	6.0	4

Table 3

Sample Problem Patients

Patient	Category	Release Time	Wait Time E	Wait Time F
1	A	0.0	4.0	6.0
2	B	1.0	3.0	5.0
3	A	2.0	2.0	4.0
4	A	3.0	1.0	3.0
5	B	3.5	0.5	2.5
6	A	4.0	0.0	2.0
7	B	4.5	N/A	1.5
8	B	5.0	N/A	1.0
9	A	5.0	N/A	1.0
10	A	5.5	N/A	0.5

Table 4

Sample Problem Airports

Airport	Capacity Category A	Capacity Category B
G	3	1
H	3	6

The problem was designed so that all the patients could not be evacuated by the fleet of aircraft. Also, the capacity of the airports was designed so that one airport could not accept all of the patients.

The solution of this problem is shown in Table 5:

Table 5

Sample Problem Solution

AIRCRAFT	ASSIGNED PATIENTS	AIRPORT ASSIGNMENT
E	1,2,4,5,6	H
F	3,8,9,10	G

The results indicate that neither the airport capacities or the aircraft capacities were exceeded. The only patient not assigned to an aircraft was #7, and this was due to the aircraft capacity. For the nine evacuated patients, the total wait time was 15 days.

3.3 Applicability to Larger Problems.

Following the solution of the small aeromedical evacuation problem, an analysis was done to determine if the methodology could be applied to solve larger problems.

The primary factor affecting the solution of the aeromedical evacuation problem, and most integer programming problems, is the number of integer decision variables. In the general formulation of the problem, the maximum number of x_{ij} variables is obtained by multiplying the number of patients by the number of aircraft. It should be noted that this is an upper bound on the number of x_{ij} variables as not every patient-aircraft combination is possible (a patient having a release time after the arrival time of an aircraft cannot be assigned to that aircraft). In the example problem with 10 patients and two aircraft, there are a total of 20 possible x_{ij} variables. However, since there were four patients whose release times were after the arrival time of the first aircraft, only 16 x_{ij} variables were necessary to model the sample problem. However, in the worst case scenario, the number of x_{ij} variables is the number of patients times the number of aircraft. The number of y_{jh} variables is the number of aircraft multiplied by the number of airports.

The total number of binary variables required to solve the problem is the number of x_{ij} variables plus the number

of y_{jh} variables. The z_{ijh} variables in this formulation are continuous and do not represent a significant computational burden to the problem.

Casualty data has been obtained from Headquarters AMC detailing the total number of causalities expected during wartime. There were three scenarios for which AMC had obtained casualty estimates. The worst case scenario was chosen in order to determine the maximum size the problem formulation could attain. The data obtained gives only the total number of expected causalities, and does not detail injury by category. Historical percentages of injuries by category are listed in Chapter 1 and do not influence the size of the problem.

In the selected scenario, the worst case generates approximately 2000 casualties per day. If there are two COMMZ evacuation points in this scenario (18), this reduces the number of patients per evacuation point to 1000. It is assumed that four aircraft will be unavailable due to maintenance, thereby reducing the available fleet to 40 aircraft. It is also assumed that one-half of the incoming Boeing 767s (20 aircraft) would arrive at a single evacuation point every two days. Therefore, approximately 10 aircraft per day arrive at the COMMZ evacuation hospital.

Based on this data, an estimate of the number of binary variables can be calculated for a given day. The

number of x_{ij} variables is 1000 patients times 10 aircraft for a total of 10,000 variables per day. The number of y_{jh} variables is significantly less than the number of x_{ij} variables. There are at most 15 airports in the model. Therefore the maximum number of y_{jh} variables is 150 per day or 10 aircraft times 15 airports. The total number of binary variables is the sum of the number of x_{ij} and y_{jh} variables and is 10,150 per day.

Problems of this size cannot be solved easily. Nemhauser and Woolsey state in their book, *Integer and Combinatorial Optimization*, that problems having several thousand integer variables have been solved optimally (13:16). In general, IP programs are classified as NP-Hard, and although some large IP problems have been solved optimally by taking advantage of special structure in the problem, an approach to take advantage of any special structure within this problem was not found. The worst case problem size is too large to expect to solve quickly in the dynamic environment which would exist in wartime; therefore, a heuristic approach has been developed and is discussed in Chapter 4.

4. *HEURISTIC DEVELOPMENT AND TESTING*

The purpose of this chapter is to describe a heuristic algorithm developed for the aeromedical evacuation problem. Following the description of the heuristic, the computational testing of the procedure will be discussed.

There are two distinct subproblems of the aeromedical evacuation problem: assigning aircraft to airports and assigning patients to aircraft. The problem was divided into two subproblems and these are solved in an iterative procedure. The first subproblem is that of assigning aircraft to airports and will be referred to as the aircraft-airport assignment problem. These results (aircraft to airport assignments) are then used as input to the second subproblem: assigning patients to aircraft. The second subproblem will be referred to as the patient-aircraft assignment problem. These subproblems are solved repeatedly until a suitable termination rule is satisfied. A flow chart of the program is shown on the next page.

4.1 Aircraft-Airport Assignment Problem.

The first subproblem is the assignment of aircraft to airports. Given an assignment of patients to aircraft

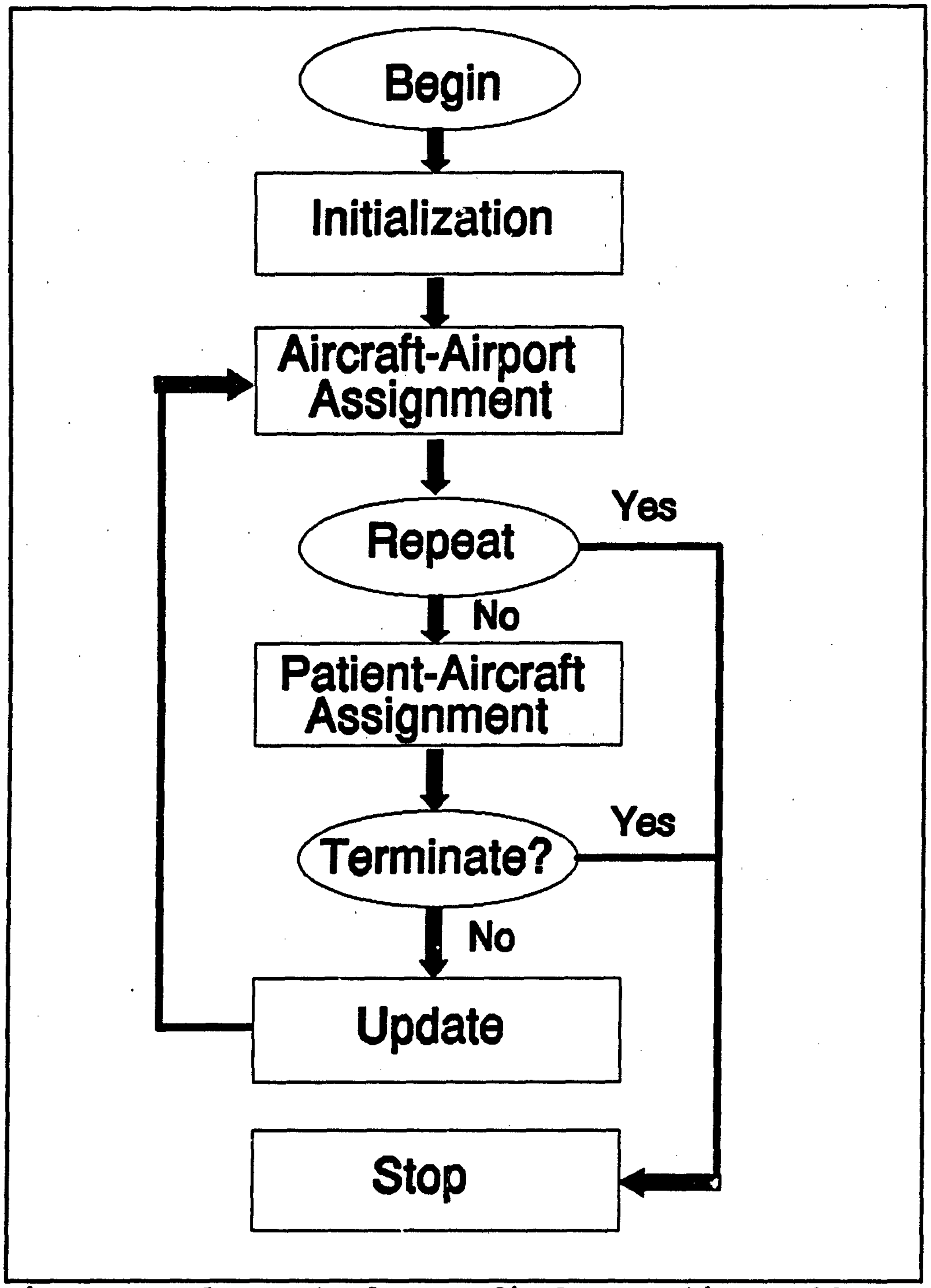

Figure 1: Flow Cart of Aeromedical Evacuation Problem

(either through an initialization procedure or from the output of the patient-aircraft assignment problem, both described later), the task is to assign aircraft to airports. Note that a feasible assignment based on hospital capacities may not be possible. Therefore, the objective of this subproblem is to minimize this infeasibility. This is accomplished through a goal programming type approach.

In the analysis of the aeromedical evacuation problem presented in Chapter 3, it was determined that the number of binary variables required for the aircraft-airport assignment problem was small enough to permit solution by integer programming. The purpose of the aircraft-airport assignment problem is to select the best subset of airports based on the distribution of patients released for evacuation to the CONUS. In this subproblem, the wait time of the patients is not a factor in the decision on where to assign the aircraft. For this subproblem we are only interested in minimizing the amount that hospital capacities are exceeded by this assignment. A goal programming type approach using deviational variables is used to ensure this.

The indexes, variables, and parameters used previously remain the same for the aircraft-airport assignment problem. Two new variables (deviational variables) and one new parameter are introduced in the formulation of the aircraft-

airport assignment problem. The new variables are as follows:

$SPLUS_{hk}$: Surplus capacity at airport h for patients in category k. This variable is the number of category k patients under the capacity of airport h.

$EXCS_{hk}$: Excess number of category k patients traveling to airport h. This variable is the number of category k patients over the capacity of airport h.

The y_{jh} variable previously defined is:

$$y_{jh} = \begin{cases} 1 \text{ if aircraft } j \text{ is assigned to airport } h \\ 0 \text{ otherwise} \end{cases}$$

The new parameter is as follows:

A_{jk}: Number of category k patients assigned to aircraft j.

The previously defined parameter is:

CH_{hk}: Hospital bed capacity of airport h for category k patients.

The IP formulation is as follows:

$$MIN \quad \sum_{h\in H} \sum_{k\in K} \quad EXCS_{hk} \tag{7}$$

SUBJECT TO:

$$\sum_{j\in J} A_{jk}\, y_{jh} + SPLUS_{hk} - EXCS_{hk} = CH_{hk} \qquad \forall\, h\in H,\ \forall\, k\in K \tag{8}$$

$$\sum_{h \in H} y_{jh} = 1 \qquad \forall j \in J \tag{9}$$

$$y_{jh} \in \{0,1\}, \quad SPLUS_{hk}, EXCS_{hk} \geq 0$$

Objective Function. The objective function minimizes the number of patients assigned to an airport in which the capacity of that airport by patient category has been exceeded. For instance, if 12 general medical category patients are assigned to aircraft j travelling to airport h where the capacity is 10 general medical patients, the value of $EXCS_{hk}$ would be two. However, if the capacity of that airport were 14, $EXCS_{hk}$ would be zero and $SPLUS_{hk}$ would have a value of two.

Constraint Sets. The first constraint set (Equation (8)) represents the bulk of the integer program. It applies a goal programming approach in which there are variables for falling short of the goal and variables for exceeding the goal. The goal in the aircraft-airport assignment problem is to minimize the number of patients assigned to an airport where a hospital bed is not available for that patient. Minimizingthe sum of the variables $EXCS_{hk}$ models this goal.

The A_{jk} parameter is the number of category k patients assigned to aircraft j. Initially, the loading of patients

is FCFS and the distribution is listed in Table 6 in Section 4.3. In subsequent iterations of the algorithm, the A_{jk} parameters are updated based on the solution of the patient-aircraft assignment problem. The y_{jh} variable is the same as in Chapter 3 and represents the assignment of aircraft j to airport h when $y_{jh} = 1$. The $SPLUS_{hk}$ variable represents the unused capacity at airport h for category k patients. The right-hand-side value (CH_{hk}) of the equality constraints is the number of category k patients that can be treated at hospital facilities served by airport h.

The second constraint set (Equation (9)) ensures that an aircraft can only be assigned to one airport. This models the assumption that each aircraft offloads it patients at a single airport.

The values of the y_{jh} variables (assignment of aircraft to airports) are passed to the patient-aircraft assignment problem.

The IP was formulated using the General Algebraic Modeling System (GAMS) and solved using the GAMS/ZOOM MIP solver on a VAX mainframe computer. The GAMS code is listed in Appendix C.

4.2 Patient-Aircraft Assignment Problem.

The second subproblem solved is the patient-aircraft assignment problem. Given a set of aircraft-airport assign-

ments, the goal is to assign patients to aircraft so that patient wait time is minimized. It would be preferable to load all aircraft to capacity, although this may not be possible due to airport capacities.

The problem is formulated similar to the formulation in Chapter 3. The major difference is that the y_{jh} variables are now fixed parameters and the z_{ijh} variables are no longer needed. The formulation is:

$$Min \sum_{i\in P} \sum_{j\in J_i} ((Arr_j - Rel_i)x_{ij} + M*x_{iD}) \qquad (1)$$

SUBJECT TO:

$$\sum_{j\in J_i} x_{ij} + x_{iD} = 1 \qquad \forall\, i\in P \qquad (2)$$

$$\sum_{i\in P} x_{ij} \le CA_j \qquad \forall\, j\in J_i \qquad (4)$$

$$\sum_{i\in P_k} \sum_{j\in J_i} y_{jh}\, x_{ij} \le CH_{hk} \qquad \forall\, h\in H, \forall\, k\in K \qquad (10)$$

$x_{ij} \in \{0,1\}$, $x_{iD} \in \{0,1\}$, for every i, for every j

Equations (1), (2), and (4) are identical to the constraints in Chapter 3. Equation (10) is an altered version of the

constraints in Chapter 3. They were altered by replacing the z_{ijh} variable with the $y_{jh}*x_{ij}$ term. The z_{ijh} variable was a product of the binary variables x_{ij} and y_{jh}, but since the aircraft-airport assignments (y_{jh}) are fixed parameters, the need for the z_{ijh} variable no longer exists.

The vast majority of the binary variables used in the formulation of the aeromedical evacuation problem presented above and in Chapter 3 were in the assignment of patients to aircraft. Therefore, a more efficient method needs to be used in order to reduce the computational time required to solve this subproblem. The method used is network flow programming. A minimum cost flow network was designed to minimize the waiting time of the patients. The network is illustrated in Figure 2. The network is a smaller version of the actual one used in the research. The network in Figure 2 uses only three categories and two aircraft. The network starts with a source node (S) which generates the number of patients released on a daily basis. There are three arcs emanating from the source node. Each arc travels to a node which represents the injury category that the patient is in and the day the patient was generated (GM1, P1, S1). The 1 in the label GM1 represents patients generated on day 1. Once again, historical percentages were used in the generation of patients. The cost along the arcs from the source node (labeled 1 in Figure 2) is zero and the

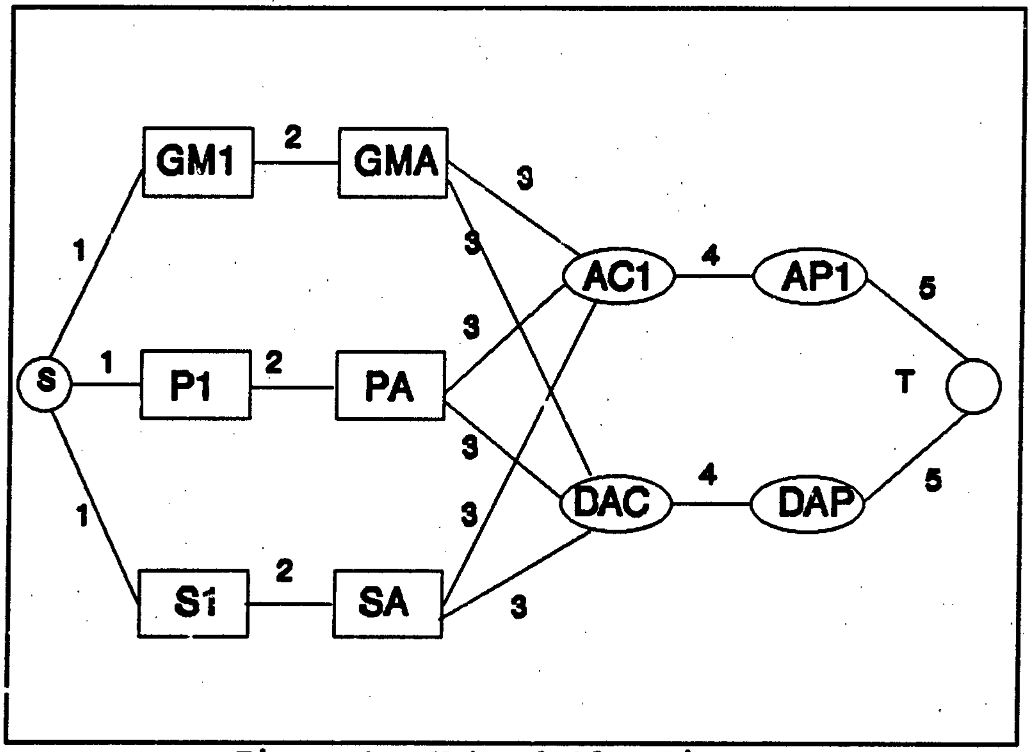

Figure 2: Network Flow Diagram

capacity is the number of patients generated for that category. From each injury category, an arc travels to a secondary injury category node (GMA, PA, SA). These arcs (labeled 2 in Figure 2) have a capacity of the number of patients in that injury category and a cost reflecting the number of days each patient has been waiting. If a patient was released on the day that the aircraft is scheduled to arrive, the cost along the arc would be zero. If the patient was generated the day before and left unassigned from the previous day, the cost would be one day. From each of the second injury category nodes, arcs travel to each air-

craft including a dummy aircraft (AC1, DAC). These arcs (labeled 3 in Figure 2) represent the assignment of patients to aircraft. The arcs' capacities are the injury category capacity of the airport to which the aircraft has been assigned. The cost on the arc is the patient's wait time. The additional wait time along this arc is the time from midnight on the current day to the arrival time of the individual aircraft. The wait time of the dummy aircraft (DAC) was set at a level much greater than the wait time of all the patients, thus discouraging the assignment of patients to the dummy aircraft. The capacity of the arcs travelling to the dummy aircraft is unlimited. From each of the aircraft nodes (including the dummy aircraft node), an arc (labeled 4 in Figure 2) travels to an airport node (AP1, DAP). The capacities of these arcs (other than those incident to the dummy aircraft) is the capacity of a Boeing 767: 100 patients. The cost of the arc is zero. A single arc (labeled 5 in Figure 2) travels from each airport node to a sink node (T) whose demand is the number of released patients in the system.

A modification to the network is required when more than one aircraft has been assigned to the same airport. The modification was needed because an airport's capacity might be exceeded by sending multiple aircraft to an airport under the present network configuration. For example,

consider a case where two aircraft have been assigned to the same airport whose GM capacity is 30. The example is illustrated in the following figure.

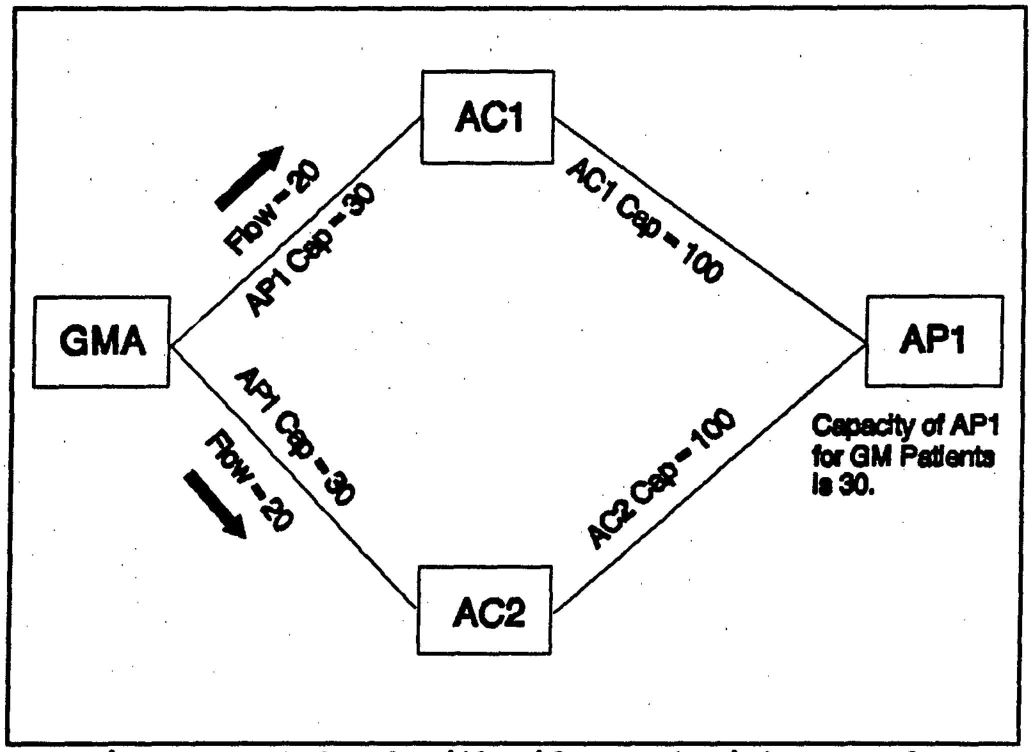

Figure 3: Network with Side Constraints Example

There are arcs travelling from the second injury category nodes to two aircraft nodes. The flow along each arc is 20 GM patients. The capacity of each of these arcs would be the capacity of the airport which is 30 GM patients. Then, there would be 40 GM patients arriving at the airport whose capacity is 30. This problem is solved by incorporating side constraints. The side constraints ensure the total arc flow out of the injury category nodes travelling to aircraft

nodes (assigned to the same airport) must be less than or equal to the capacity of the airport by injury category.

The network was formulated and solved using the SAS/OR package on a VAX mainframe computer. The SAS code is listed in Appendix D. The results of the minimum cost flow program were then used as input to the aircraft-airport assignment problem.

4.3 Aeromedical Evacuation Heuristic Algorithm.

The two assignment problems were linked in one algorithm. In order to begin the algorithm there must be an initial assignment of patients to aircraft. Since the overall goal of the heuristic algorithm is to minimize the patients' waiting time, the obvious choice for the initial loading is to evacuate the patient as soon as possible after he/she is released. Therefore, the dispatching rule used is first come first serve (FCFS). The aircraft-airport assignment problem is solved first to take advantage of this easy initialization scheme. The patient-aircraft assignment problem is then solved using the generated aircraft to airport assignments. If necessary, the process is repeated until one of the termination criteria is met.

The algorithm is as follows:

AEROMEDICAL EVACUATION HEURISTIC ALGORITHM

Step 0. Initialization. Assign patients to aircraft on a first come first serve basis based on release times. Initialize iteration counter to 0.

Step 1. Aircraft-Airport Assignment. Given patient-aircraft assignments, assign aircraft to airports. Pass aircraft-airport assignments to Step 2. If assignments are repeated between iterations, proceed to Step 3. Increment iteration counter by 1.

Step 2. Patient-Aircraft Assignment. Given a set of aircraft-airport assignments, solve the minimum cost flow problem minimizing patient wait time. Proceed to Step 3.

Step 3. Check for termination. Proceed to Step 5 if:
1) Aircraft loaded to capacity.
2) Aircraft-airport assignments remained unchanged between iterations.
3) User prescribed iteration limit exceeded.

If not proceed to Step 4.

Step 4. Update. Update A_{jk} to the aircraft loads determined by Step 2. Proceed to Step 1.

Step 5. Stop. A feasible assignment of aircraft to airports and patients to aircraft has been found.

The first step of the algorithm (Step 0) is the initialization step. All aircraft are loaded with patients being assigned to aircraft using a FCFS dispatching rule. The parameter A_{jk} representing the patient-aircraft assignments, is passed to the aircraft-airport assignment problem.

Step 1 is the aircraft-airport assignment problem. This problem is solved and the results (aircraft-airport assignments) are passed to the patient-aircraft assignment problem. Even if the objective function value is zero, the aircraft-airport assignments still need to be passed to the patient-aircraft assignment problem. This is necessary to obtain the wait time of the patients. If the aircraft-airport assignments are repeated between iterations, the algorithm proceeds to Step 3 and terminates. Finally, the iteration counter is incremented by 1 before proceeding to Step 2.

Step 2 is the patient-aircraft assignment problem. There are two results from the patient-aircraft assignment problem. The results are the new patient-aircraft assignments (A_{jk}) and the total patient wait time.

The results are passed to Step 3. At Step 3, the results of Step 1 and Step 2 are used to determine if the algorithm can be terminated. If the results indicate that all aircraft (excluding the dummy aircraft) are loaded to capacity, the algorithm will proceed to Step 5 and termi-

nate. If all aircraft are loaded to capacity, there are no airport capacities that are violated. This is due to the design of the network. The capacities are not violated because the network will not allow any flow greater than the airport capacity. Instead, the network will assign patients to the dummy aircraft. Also, the patient wait time is minimized. This is the criteria that is desirable to terminate from, because the solution is optimal for the selected aircraft-airport assignments and the distribution of patients. The algorithm also proceeds to Step 5 and terminates if a prescribed iteration limit is exceeded or if the aircraft-airport assignments are unchanged from previous iterations. Since the aeromedical evacuation problem needs to be solved frequently and quickly, an iteration limit must be set. If the aircraft-airport assignments are repeated and the algorithm is not terminated, the algorithm would be in an infinite loop.

If any aircraft is not loaded to capacity, the algorithm proceeds to Step 4. At Step 4 the patient-aircraft assignments (new A_{jk}) are updated and they are passed to the aircraft-airport assignment problem. Step 4 readjusts the aircraft loads for input into the aircraft-airport assignment problem. This step essentially reloads each aircraft so that the total patient wait time is minimized for the current aircraft-airport assignments. The algorithm then

proceeds to Step 1 with the new values for the A_{jk} parameter, and the aircraft-airport assignment problem is re-optimized based on the new distribution of aircraft loads. At Step 1, the goal is to pick a new set of airports to assign the aircraft to. The algorithm proceeds as described above until one of the termination criteria is met.

4.4 Algorithm Testing.

The algorithm was initially tested using three separate data sets which represented realistic patient loads based on historical percentages. A subset of 13 airports was selected which would exercise the algorithm. The remaining airports were not considered by the aircraft-airport assignment problem. The subset of airports is listed in Appendix E. The subset was used to keep the number of binary variables at a manageable level (approximately 130) and to keep computational time reasonable. Several of the airports selected did not have a capacity to handle certain categories of patients, most notably the burn and spinal categories. The number of patients by category for FCFS is assumed to follow the historical percentages listed previously in Chapter 1. Initially, all of the aircraft will be loaded with the same number and same type of patients. Since the assumed aircraft capacity is 100 for each type of aircraft, the patient load per aircraft is as follows:

Table 6

Initial Patient Load

Patient Category	Number
General Medical	13
Psychological	3
Surgical	44
Orthopedic	36
Burn	3
Spinal	1

Since the FCFS load would have some of these category patients on board each aircraft, there would be a non-zero objective function value in the aircraft-airport assignment in the first iteration. At Step 2, the patient-aircraft assignment problem flow would then reassign patients based on the capacities of the airports selected. If necessary, other iterations of the algorithm would be performed until all patients were assigned to an available bed. The initial number of patients for the three data sets are listed in the following table:

Table 7

Patient Data Set Distribution

Category	Set 1	Set 2	Set 3
Gen Medical	130	130	143
Psychological	30	30	33
Surgical	440	440	484
Orthopedic	360	360	396
Burn	30	20	33
Spinal	10	20	11

The first test of the algorithm was performed using data set 1. There were 1000 patients which were all generated at time 0. There were 10 aircraft whose arrival times were spaced out evenly at 0.1 day intervals throughout the day. The penalty for an assignment of a patient to the dummy aircraft in the patient-aircraft assignment problem was set to 100. This value is much greater than the planning horizon (1 in this data set). The results of the aircraft-airport assignment problem for data set 1 are listed in Appendix F. The results are summarized in the following table:

Table 8

Testing of Data Set 1

Iter	Capacities Violated at Step 1	Patients Assigned to Dummy AC (Step2)	Wait Time
1	19	4	846.4
2	4	1	549.1
3	1	0	450

The results indicate that, on the first iteration, 19 patients have violated the hospital capacity constraints. The airport assignments were input to the patient-aircraft assignment and the patients were reassigned to minimize their wait times. This resulted in a new assignment of patients to aircraft which was different than the assignments in the initialization step. In the patient-aircraft assignment problem, the number of patients violating hospital capacity constraints was reduced from 19 to 4 as indicated by the assignment of four burn patients to the dummy aircraft. The total wait time was 846.4 days of which 400 days were due to the patients assigned to the dummy aircraft. The first nine aircraft were loaded to capacity and the average wait time for the assigned patients was 0.448 days per patient. The aircraft loading was changed dramatically and is listed in Appendix F. The new aircraft loading was input to the aircraft-airport assignment problem. The

results indicate that the objective value remained at four, but a different assignment of aircraft to airports was made. Inputting the results into the patient-aircraft assignment problem reduced the number of patients violating hospital capacity constraints to one. The total wait time was 549.1 days, of which 100 days were due to a single patient assigned to the dummy aircraft. The first nine aircraft were once again loaded to capacity and the average wait time was 0.449 days per assigned patient. The reason the average wait time went up is that three more patients were assigned to the aircraft with the latest arrival time. Once again the aircraft loads were changed and a third aircraft-airport assignment problem was solved. The objective value remained at one, but a new assignment of aircraft to airport assignments was made. These new assignments were input into the patient-airport assignment problem and the result was that all patients were assigned to Boeing 767 aircraft. The wait time was 450 days, for an average of 0.45 days per patient. The 0.45 days per patient is consistent for the data set, since the average arrival time for the aircraft was 0.45 days.

Three complete iterations were performed by the algorithm and at each step the number of patients violating hospital capacity constraints either was reduced or stayed the same.

The second data set required only one complete iteration of the algorithm. There were 1000 patients in data set 2, but in a different distribution of injury types than in data set 1. The arrival of aircraft remained the same as data set 1. In the initial aircraft-airport assignment problem, 18 patients violated airport capacity constraints. However, when the assignments of aircraft to airports were input into the patient-aircraft assignment problem, the aircraft loads were readjusted and no patient was assigned to the dummy aircraft. Once again the total wait time was 450 days, for an average per patient of 0.45 days. The results of the iteration are listed in Appendix G. The results of data Sets 2 and 3 are summarized in the following table. The DAC Asmts column is the assignment of patients to the dummy aircraft.

Table 9

Violated Capacities for Data Sets 2 and 3

Data Set	Step 1	Step 2	DAC Asmts	Wait
2	18	0	0	450
3	27	0	100	10,463

The third data set involved generating and scheduling patients over a two day period. There were 100 patients remaining from the previous day, which simulates that there was an insufficient number of aircraft to evacuate all the

patients on the previous day. It was assumed that the distribution of patients remaining overnight followed the historical distribution. The aircraft schedule remains unchanged from the two previous test cases. On the second day, 1000 patients were generated and the distribution of patients was identical to that of data set 1. The purpose of designing this data set is to determine if the algorithm would transport patients in the order they are generated. In other words, patients generated on the first day should be evacuated before the patients generated on the second day.

In the initialization step, each of the 10 aircraft was assigned 110 patients. Exceeding the capacity of each aircraft is acceptable in the airport-aircraft assignment subproblem. The important factor is obtaining optimal aircraft-airport assignments. Exceeding the capacity of the aircraft is not important. The number and distribution of patients is the information that the aircraft-airport assignment problem needs in order to optimally solve the problem.

The results of the iteration are listed in Appendix H. The objective function value indicated that 27 patients had violated airport capacity constraints. However, since the total capacity of the 10 aircraft is 1000, in the patient-aircraft assignment problem there will be at least 100

patients that will be assigned to the dummy aircraft. The airport assignments were passed to the patient-aircraft assignment problem and the patient-aircraft problem was solved. The results indicated that 100 patients were assigned to the dummy aircraft. This indicated that the 10 actual Boeing 767 aircraft were filled to capacity. Of the 100 patients assigned to the dummy aircraft, 81 patients were from the orthopedic category and 19 were from the burn category. All of these 100 patients were generated on the second day. The total wait time was 10,463 days, of which 10,000 was from the patients assigned to the dummy aircraft. The average wait time for the patients not assigned to the dummy aircraft was 0.463 days. The reason that the average wait time went up to 0.463 days from 0.45 days is that the wait time for all of the 100 patients from the first day is greater than one day. The significance of this iteration is that patients would be airlifted as soon as an aircraft was available, and patients would not be left for an extended period of time waiting to be airlifted.

4.5 Analysis of Results.

The results of the testing of the Aeromedical Evacuation Heuristic Algorithm indicated that the method developed can be used by medical regulators to determine who goes on what aircraft and where to send the aircraft. This method

could be used whether the present hub and spoke system is maintained or if an alternative system such as direct delivery is implemented. The method can reduce the amount of stateside redistribution of patients, saving valuable resources. There are several considerations that merit discussion.

While the method did work within several iterations on the tested data sets, it does not guarantee an optimal solution will be obtained in all cases. Obviously, if there are more patients than there are available hospital beds, aeromedical evacuation of all patients would be impossible. It is possible that the solution to the aircraft-airport assignment problem will be a set of airports in which there will be some patients violating capacity constraints. The patient-aircraft assignment program may readjust the aircraft loads and pass those along for input into the aircraft-airport assignment problem. It is possible that the aircraft-airport assignment program may then pick the same subset of airports as the previous iteration. The algorithm guarantees that the solution between steps of the algorithm will not get worse. However, the algorithm does not guarantee the next solution will get better. Also, there does not appear to be a bound on how close to optimality the current solution is.

There are several options that the regulator can choose from if this situation occurs. The first is that they may pick a new group of airports (different from the initial set) for the aircraft-airport assignment program to consider and start the process over again. The second option is that the regulator may choose to not fully load an aircraft. The third and final option is for the regulator to reassign the aircraft to an airport with redistribution capability.

Another concern is the computational time required when executing the algorithm. Since the first subproblem is an IP, the computational time required to solve the problem cannot be ignored. The computational times for the IP using the GAMS/ZOOM solver varied from a few seconds to three hours. The problem had 130 binary variables, 297 continuous variables and 90 constraints. The GAMS/ZOOM solver uses a combination Linear Program, Pivot and Complement heuristic, and Branch and Bound procedure. It first solves a relaxed LP to get a lower bound. It then uses the Pivot and Complement heuristic to come up with an integer answer. If the heuristic fails to come up with the optimal solution, GAMS/ZOOM will then use Branch and Bound (7:226-227). In problems that took a long time to solve, the solver was searching nodes of the branch and bound tree to reach the optimal answer. Doing branch and bound generally takes far

longer than the initial LP and the Pivot and Complement heuristic combined (7:227).

The third concern is the amount of time required for inputting the data. The implementation of this algorithm used two completely different systems to solve problems (GAMS and SAS). There is no direct link between the two systems at AFIT. Therefore, much of the data between iterations had to be manually entered. For the aircraft-airport assignment problem using GAMS, the A_{jk} parameter had to be changed between iterations (See Appendix C: TABLE A(J,K)). In the patient-aircraft assignment problem using SAS, an entire column of the SAS data set must be manually changed (See Appendix D: _capac_ column). The time required for these inputs is not excessive, but their entry would waste valuable time in an actual wartime environment.

Another possible concern is the side constraints in the patient-aircraft assignment network problem. A pure network problem can be solved using a polynomial time algorithm, while solving a problem with side constraints cannot be. It is theoretically possible that solving the network with side constraints might take an excessive length of time. However, no problems of this nature were encountered in any of the SAS runs with side constraints. The computational time required on all runs was under 30 seconds. A pure network

was designed which would delete the need for side constraints. The network is illustrated in Figure 4.

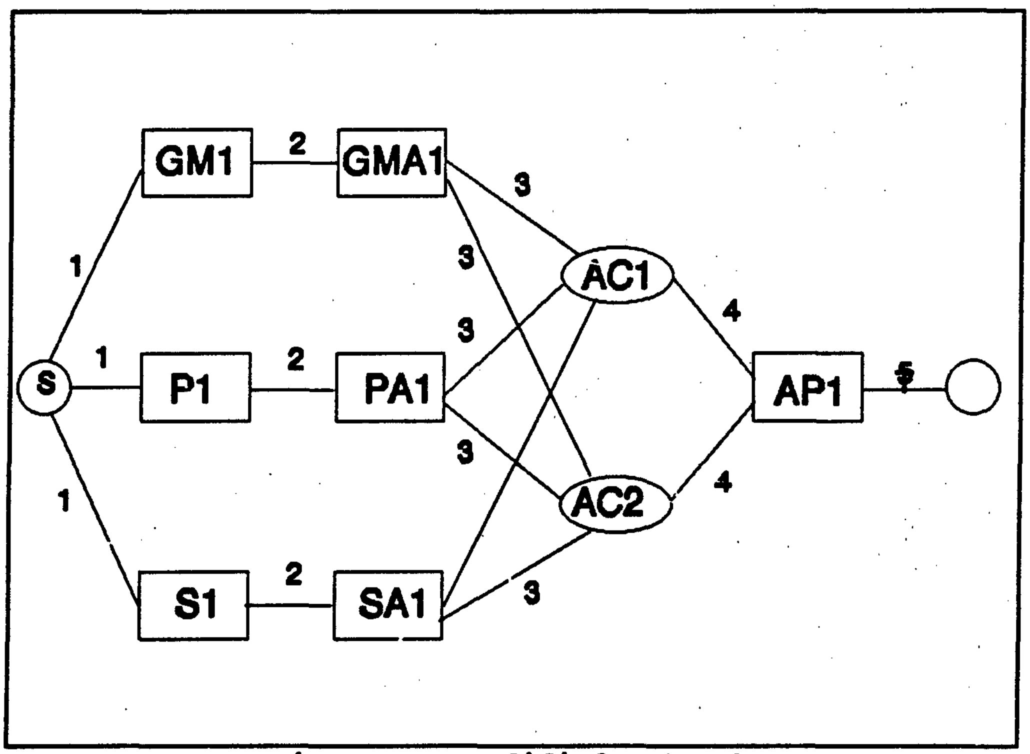

Figure 4: Modified Network

Initially, the network is identical to the network used in the research. There are three arcs emanating from the source (S) which travel to injury category nodes (Figure 4: GM1, P1, S1). The cost and capacities are the same as the original network used in the research. However, this is where the similarity ends. From each injury category node there are arcs travelling to injury category nodes for each airport (GMA1, PA1, SA1). There are six injury category nodes for each airport. The capacity of the arcs (labeled 2 in Figure 4) is the injury category capacity of the airport

to which the aircraft (or multiple aircraft) has been assigned. The cost is the wait time of the patients in that category. This is the modification to the network that was made so that the side constraints can be deleted. From these nodes, there are arcs travelling to aircraft nodes (Figure 4: AC1, AC2). The cost along these arcs is zero, and the capacity is the airport's capacity for that injury category. From each aircraft node, a single arc travels to an airport node (Figure 4: AP1). The cost along this arc is zero and the capacity is the capacity of a Boeing 767; 100. From the airport node a single arc travels to the sink node. The cost along the arc is zero and the capacity is the total capacity of the airport (all categories combined). The size of the network might be a concern. The network as formulated this way is approximately 4 times larger than the present network. The number of nodes increases from 58 in the network that was used in the research to 319 in the proposed network and the number of arcs increases from 245 to 831. It is larger because each aircraft requires six injury category nodes, whereas the smaller network requires six injury category nodes for <u>all</u> aircraft regardless of destination.

5. CONCLUSIONS AND RECOMMENDATIONS

The purpose of this chapter is to summarize the research effort, to discuss the major conclusions, and to provide recommendations for further research.

5.1 Summary and Conclusions

This thesis effort examined the aeromedical evacuation process and provided a method of scheduling patients on and routing CRAF Boeing 767 aircraft. Presently, the Boeing 767 travel from COMMZ evacuation airports to major hubs in the CONUS. At the major hubs, the patients are unloaded off the Boeing 767 and loaded onto C-130 aircraft which then transport the patients from the hubs to the airport nearest the patient's assigned hospital bed. This research has examined and presented a method to schedule patients on specific Boeing 767 aircraft and then route the aircraft to a single CONUS airport. This method of scheduling and routing can be implemented within the present system or it can be used in alternative systems such as a direct delivery system.

The aeromedical evacuation problem that was studied had the objective of minimizing patient wait time in the theater of conflict. By minimizing the wait time, the patients would benefit from the improved medical care that is available at the CONUS hospital. The problem was modeled as a

mixed integer program (MIP). The problem had two primary assignment variables: assigning aircraft to airports and assigning patients to aircraft. The problem was constrained by two capacities. Aircraft capacity is the capacity of a Boeing 767. Airport capacity was defined to be the number of patients in each injury category that could be served by the hospitals served by that airport.

A small problem was formulated, solved, and an analysis was performed on the structure of the problem. The analysis revealed that the number of binary variables required to formulate a realistic problem was too large to be solved efficiently. Since the aeromedical evacuation needs to be solved quickly and frequently, a heuristic procedure was developed.

The heuristic procedure divided the problem into two subproblems. The first step was to initially load aircraft with patients on a first come first serve basis. The first subproblem that was solved was the assignment of aircraft to airports. This assignment problem was solved by integer programming using techniques borrowed from goal programming. Deviational variables were used to minimize the number of patients travelling to an airport whose surrounding hospitals could not support them. The results of the IP were a set of aircraft-airport assignments. These assignments and

their associated capacities were then passed to the second subproblem.

The second subproblem was the patient to aircraft assignment problem. In the original problem formulation, the vast majority of the binary variables used to formulate the problem were in this assignment. Network flow programming was used in order be more efficient in the solution process. If all of the aircraft were loaded to capacity, meaning that the airport capacity constraints were not violated, the algorithm was terminated. The algorithm was terminated because optimal assignments of both subproblems have been reached. If not, the new patient-aircraft assignments were passed to the aircraft-airport assignment subproblem. The algorithm proceeded in this manner until one of the termination criteria was met.

The algorithm was tested using three different but realistic data sets. The results indicate that this method of scheduling patients and routing aircraft was feasible. This method could be used in the present hub and spoke system and in alternative systems. In the testing of the first data set, three iterations of the algorithm were performed before no airport capacities were violated. The remaining two data sets required only one iteration of the algorithm.

5.2 Recommendations for Further Study

In conducting the research, five areas were uncovered upon which further study and research could be performed.

The first area is to put the algorithm in a dynamic environment. The development and testing of the algorithm was done in a static environment. By placing the algorithm into a dynamic test environment, one would be able to determine the effects that the routing and scheduling algorithm has upon patient flow, aircraft flow, and hospital capacities. One possibility is to incorporate the method into Charles Wolfe's SIMSCRIPT simulation model developed in his AFIT thesis titled "The Use of Simulation to Evaluate Strategic Aeromedical Evacuation Policy and Planning" (21).

The second recommendation is to test the impact of the length of the planning horizon upon this method of patient scheduling and aircraft routing. In the model used in this research, the number and type of patients left unassigned had no effect on the next days decision process. The planning horizon's length may have an effect on what type and how many patients are not assigned to an aircraft. The planning horizon's length may also have an effect on the penalty applied to the dummy aircraft in both the small aeromedical evacuation problem and to the patient-aircraft network assignment problem.

The third recommendation is to study and implement the pure network formulation of the patient-aircraft assignment problem. The solution time to the problem as formulated using side constraints could prove to be excessive. Would the pure network, which is four times larger than the present network, solve in a shorter time than the present network with side constraints?

The fourth recommendation is to investigate more efficient methods to solve the aircraft-airport assignment problem. The solution time of the present IP formulation can be excessive given the dynamic nature of the wartime aeromedical evacuation process. One possible method would be to formulate the aircraft-airport assignment as a network problem which would reduce the computational time required to solve the problem.

The fifth and final recommendation is to integrate the algorithm's two main subproblems into one programming language. This would decrease the manual inputs required. As a goal, the algorithm should be totally automated so the problem could be solved quickly and efficiently. The present algorithm uses two different languages (GAMS and SAS), which are incompatible. Such an implementation would be necessary for this approach to be acceptable as an efficient planning tool.

Appendix A: Hospital Capacities

AIRPORT	GEN MED	PYSCH	SURG	ORTH	SPIN	BURN
LOS ANGELES	4704	1343	3327	1102	567	473
TUSCON	169	66	82	35	9	8
LUKE	371	110	472	47	21	5
SAN FRAN	897	421	1481	358	117	56
FT LEWIS	473	180	450	264	81	77
PORTLAND	213	64	327	81	15	12
BOSTON	3593	303	2612	368	97	55
NHAMPTON	233	6	132	66	4	5
ALBANY	130	104	181	103	1	1
CHARLOTTE	527	344	370	102	41	14
FT JACKSON	420	110	242	149	24	27
FT GORDON	274	212	169	150	29	14
FT BRAGG	1621	533	1632	714	142	135
CHARLESTON	212	78	277	62	3	12
DENVER	603	190	732	437	26	3
HILL	171	72	186	32	20	17
WICHITA	52	30	46	42	0	1
ALBQ	118	27	123	36	0	5
FT BLISS	603	115	539	389	27	3
PHIL	4209	1286	3367	784	228	102
SYRACUSE	219	31	47	16	4	38
BUFFALO	813	128	671	202	25	17
PITTSBURG	1366	110	948	134	30	10
NORFOLK	1201	516	1061	434	22	46
WASH DC	1604	357	1403	711	170	23
HOUSTON	1211	246	1205	402	5	26
NEW ORLEANS	2323	635	1271	325	79	23
LITTLE ROCK	80	10	83	86	0	0
SHREVEPORT	229	101	106	32	0	4
OK CITY	227	52	422	120	1	0
CARSWELL	683	147	502	174	52	0
SAN ANTONIO	1150	291	428	138	9	35

Appendix A: Hospital Capacities

AIRPORT	GEN MED	PYSCH	SURG	ORTH	SPIN	BURN
ATLANTA	706	84	467	309	3	35
BIRMINGHAM	514	71	487	56	12	9
ORLANDO	875	154	702	201	57	50
JACKSONVILLE	456	193	393	355	54	57
JACKSON	336	166	315	23	16	8
MILLINGTON	427	160	248	59	25	4
KNOXVILLE	206	57	209	64	9	1
NASHVILLE	468	199	490	161	13	5
CHICAGO	2204	568	2651	547	88	95
CLEVELAND	344	197	287	103	41	33
MINNEAPOLIS	88	149	338	73	27	11
DES MOINES	65	27	84	12	4	3
INDIANAPOLIS	250	77	84	33	8	3
SCOTT	665	130	1216	276	0	0
LEAVENWORTH	222	76	382	84	24	6
LEXINGTON	261	92	395	192	5	1
ALLEN PARK	592	154	514	130	29	26
OFFUT	583	104	126	178	14	16
WRIGHT-PATT	537	155	880	318	120	96

Appendix B: Aeromedical Problem Formulation and Solution

FORMULATION:

```
MIN      4 X1E + 6 X1F + 3 X2E + 5 X2F + 2 X3E + 4 X3F +
           X4E + 3 X4F + 0.5 X5E + 2.5 X5F + 2 X6F +
       1.5 X7F + X8F + X9F + 0.5 X10F + 10 X1D + 10 X2D +
        10 X3D + 10 X4D + 10 X5D + 10 X6D + 10 X7D +
        10 X8D+ 10 X9D + 10 X10D

SUBJECT TO:

        2)   X1E + X1F + X1D =  1
        3)   X2E + X2F + X2D =  1
        4)   X3E + X3F + X3D =  1
        5)   X4E + X4F + X4D =  1
        6)   X5E + X5F + X5D =  1
        7)   X6F + X6D + X6E =  1
        8)   X7F + X7D    =  1
        9)   X8F + X8D    =  1
       10)   X9F + X9D    =  1
       11)   X10F + X10D =  1
       12)   YEG + YEH    =  1
       13)   YFG + YFH    =  1
       14)   X1E + X2E + X3E + X4E + X5E + X6E <=    5
       15)   X1F + X2F + X3F + X4F + X5F + X6F+
             X7F + X8F + X9F + X10F <=    4
       16)   Z1EG + Z3EG + Z4EG + Z6EG + Z1FG +
             Z3FG + Z4FG + Z6FG + Z9FG + Z10FG <=    3
       17)   Z2EG + Z5EG + Z2FG + Z5FG + Z7FG + Z8FG <=    1
       18)   Z1EH + Z3EH + Z4EH + Z6EH + Z1FH +
             Z3FH + Z4FH + Z6FH + Z9FH + Z10FH <=    3
       19)   Z2EH + Z5EH + Z2FH + Z5FH + Z7FH + Z8FH <=    4
       20)   X1E + YEG - Z1EG <=    1
       21)   X2E + YEG - Z2EG <=    1
       22)   X3E + YEG - Z3EG <=    1
       23)   X4E + YEG - Z4EG <=    1
       24)   X5E + YEG - Z5EG <=    1
       25)   YEG + X6E - Z6EG <=    1
       26)   X1E + YEH - Z1EH <=    1
       27)   X2E + YEH - Z2EH <=    1
       28)   X3E + YEH - Z3EH <=    1
       29)   X4E + YEH - Z4EH <=    1
       30)   X5E + YEH - Z5EH <=    1
       31)   YEH + X6E - Z6EH <=    1
       32)   X1F + YFG - Z1FG <=    1
       33)   X2F + YFG - Z2FG <=    1
```

```
 34)    X3F + YFG - Z3FG <=    1
 35)    X4F + YFG - Z4FG <=    1
 36)    X5F + YFG - Z5FG <=    1
 37)    X6F + YFG - Z6FG <=    1
 38)    X7F + YFG - Z7FG <=    1
 39)    X8F + YFG - Z8FG <=    1
 40)    X9F + YFG - Z9FG <=    1
 41)    X10F + YFG - Z10FG <=    1
 42)    X1F + YFH - Z1FH <=    1
 43)    X2F + YFH - Z2FH <=    1
 44)    X3F + YFH - Z3FH <=    1
 45)    X4F + YFH - Z4FH <=    1
 46)    X5F + YFH - Z5FH <=    1
 47)    X6F + YFH - Z6FH <=    1
 48)    X7F + YFH - Z7FH <=    1
 49)    X8F + YFH - Z8FH <=    1
 50)    X9F + YFH - Z9FH <=    1
 51)    X10F + YFH - Z10FH <=    1
END
INTEGER       X1E
INTEGER       X1F
INTEGER       X2E
INTEGER       X2F
INTEGER       X3E
INTEGER       X3F
INTEGER       X4E
INTEGER       X4F
INTEGER       X5E
INTEGER       X5F
INTEGER       X6E
INTEGER       X6F
INTEGER       X7F
INTEGER       X8F
INTEGER       X9F
INTEGER       X10F
INTEGER       X1D
INTEGER       X2D
INTEGER       X3D
INTEGER       X4D
INTEGER       X5D
INTEGER       X6D
INTEGER       X7D
INTEGER       X8D
INTEGER       X9D
INTEGER       X10D
INTEGER       YEG
INTEGER       YEH
INTEGER       YFG
INTEGER       YFH
```

LINDO SOLUTION:

OBJECTIVE FUNCTION VALUE

1) 25.000000

VARIABLE	VALUE	REDUCED COST
X1E	1.000000	4.000000
X1F	.000000	6.000000
X2E	1.000000	3.000000
X2F	.000000	5.000000
X3E	.000000	2.000000
X3F	1.000000	4.000000
X4E	1.000000	1.000000
X4F	.000000	3.000000
X5E	1.000000	.500000
X5F	.000000	2.500000
X6E	1.000000	.000000
X6F	.000000	2.000000
X7F	.000000	1.500000
X8F	1.000000	1.000000
X9F	1.000000	1.000000
X10F	1.000000	.500000
X1D	.000000	10.000000
X2D	.000000	10.000000
X3D	.000000	10.000000
X4D	.000000	10.000000
X5D	.000000	10.000000
X6D	.000000	10.000000
X7D	1.000000	10.000000
X8D	.000000	10.000000
X9D	.000000	10.000000
X10D	.000000	10.000000
YEG	.000000	.000000
YEH	1.000000	.000000
YFG	1.000000	.000000
YFH	.000000	.000000
Z1EG	.000000	.000000
Z3EG	.000000	.000000
Z4EG	.000000	.000000
Z6EG	.000000	.000000
Z1FG	.000000	.000000
Z3FG	1.000000	.000000
Z4FG	.000000	.000000
Z6FG	.000000	.000000
Z9FG	1.000000	.000000

Z10FG	1.000000	.000000
Z2EG	.000000	.000000
Z5EG	.000000	.000000
Z2FG	.000000	.000000
Z5FG	.000000	.000000
Z7FG	.000000	.000000
Z8FG	1.000000	.000000
Z1EH	1.000000	.000000
Z3EH	.000000	.000000
Z4EH	1.000000	.000000
Z6EH	1.000000	.000000
Z1FH	.000000	.000000
Z3FH	.000000	.000000
Z4FH	.000000	.000000
Z6FH	.000000	.000000
Z9FH	.000000	.000000
Z10FH	.000000	.000000
Z2EH	1.000000	.000000
Z5EH	1.000000	.000000
Z2FH	.000000	.000000
Z5FH	.000000	.000000
Z7FH	.000000	.000000
Z8FH	.000000	.000000

NO. ITERATIONS= 128
BRANCHES= 1 DETERM.= 1.000E 0

Appendix C: GAMS Integer Program

```
SETS
    J   aircraft      /A1,A2,A3,A4,A5,A6,A7,A8,A9,A10/

    H   hubs         /CIND, CLEX, CSCO,
                      AKNX,
                      HLRF, HOKC, HSHR, HCRS,
                      CHCHS,
                      DABQ, DWIT,
                      BALB, BNRH/

    K  categories   /GM, PYS, SUR, ORTH, SP, BUR/;


    TABLE CAPAC(H,K)  capacities of hubs by categories

                         GM      PYS      SUR      ORTH      SP
BUR
          CIND           250     77       84       33        8        3
          CLEX           261     92       395      192       5        1
          CSCO           665     130      1216     276       0        0
          AKNX           206     57       209      64        9        1
          HLRF           80      10       83       86        0        0
          HOKC           227     52       422      120       1        0
          HSHR           229     101      106      32        0        4
          HCRS           683     147      502      174       52       0
          CHCHS          212     78       277      62        3        12
          DABQ           118     27       123      36        0        5
          DWIT           52      30       46       42        0        1
          BALB           180     104      181      103       1        1
          BNRH           233     6        132      66        4        5
;

TABLE  A(J,K)   load per aircraft of category k patients

                         GM      PYS      SUR      ORTH      SP
BUR
                 A1      13      3        44       36        1        3
                 A2      13      3        44       36        1        3
                 A3      13      3        44       36        1        3
                 A4      13      3        44       36        1        3
                 A5      13      3        44       36        1        3
                 A6      13      3        44       36        1        3
                 A7      13      3        44       36        1        3
                 A8      13      3        44       36        1        3
                 A9      13      3        44       36        1        3
                 A10     13      3        44       36        1        3
;
```

```
SCALARS     NOA   number of aircraft in model /10/
            SRH   right hand side value        /1/;

VARIABLES
            Y(J,H)    assignment of plane j to hub h
            SPLUS(H,K)  surplus variable
            EXCS(H,K)   excess variable
            OPT         optimal solution value;

POSITIVE VARIABLES SPLUS, EXCS;

BINARY VARIABLE Y;

EQUATIONS
          ASSN      objective function
          EHP(H,K)  hospital capacity equations
          EMNA      maximum number of aircraft
          EHTA(J)   one hub to an aircraft;

ASSN ..       OPT =E= SUM((H,K),EXCS(H,K));
EHP(H,K) ..   SUM(J,A(J,K)*Y(J,H)) + SPLUS(H,K) - EXCS(H,K)
              =E=  CAPAC(H,K);
EMNA ..       SUM((J,H),Y(J,H)) =E= NOA;
EHTA(J) ..    SUM(H,Y(J,H)) =E= SRH;

MODEL HUBASSIGN /ALL/;

OPTION OPTCR = 0.1;
OPTION LIMROW = 0;
OPTION LIMCOL = 0;
OPTION ITERLIM = 900000;
OPTION RESLIM = 10000;
OPTION WORK = 100000;

SOLVE HUBASSIGN USING MIP MINIMIZING OPT;
```

Appendix D: SAS Network Code

```
title 'Patients to Aircraft: Hubs Already Selected';

title3 'Nodes for Network';
     data noded;
     input _node_ $ _sd_;
     cards;
     S 1100
     T -1100
     ;

title3 'Arc Data';
     data arcd;
       input _from_ $ _to_ $ _cost_ _capac_ _lo_ _name_ $;
       cards;

S    GM0 0    13    . SGM0
S    P0  0    3     . SP0
S    S0  0    44    . SS0
S    O0  0    36    . SO0
S    B0  0    3     . SB0
S    SP0 0    1     . SSP0

S    GM1 0    130   . SGM1
S    P1  0    30    . SP1
S    S1  0    440   . SS1
S    O1  0    360   . SO1
S    B1  0    30    . SB1
S    SP1 0    10    . SSP1

GM0 A11 1    250  . G0A1
GM0 A12 1.1  212  . G0A2
GM0 A13 1.2  118  . G0A3
GM0 A14 1.3  261  . G0A4
GM0 A15 1.4  683  . G0A5
GM0 A16 1.5  206  . G0A6
GM0 A17 1.6  665  . G0A7
GM0 A18 1.7  233  . G0A8
GM0 A19 1.8  683  . G0A9
GM0 A10 1.9  180  . G0A0
GM0 DAC 101  5000 . G0DAC

GM1 A11 0    250  . G1A1
GM1 A12 0.1  212  . G1A2
GM1 A13 0.2  118  . G1A3
GM1 A14 0.3  161  . G1A4
GM1 A15 0.4  683  . G1A5
```

GM1	A16	0.5	206	.	G1A6
GM1	A17	0.6	665	.	G1A7
GM1	A18	0.7	233	.	G1A8
GM1	A19	0.8	683	.	G1A9
GM1	A10	0.9	180	.	G1A0
GM1	DAC	100	5000	.	G1DAC
P0	A11	0	77	.	P0A1
P0	A12	0.1	78	.	P0A2
P0	A13	0.2	27	.	P0A3
P0	A14	0.3	92	.	P0A4
P0	A15	0.4	147	.	P0A5
P0	A16	0.5	57	.	P0A6
P0	A17	0.6	130	.	P0A7
P0	A18	0.7	6	.	P0A8
P0	A19	0.8	147	.	P0A9
P0	A10	0.9	104	.	P0A0
P0	DAC	100	5000	.	P0DAC
P1	A11	0	77	.	P1A1
P1	A12	0.1	78	.	P1A2
P1	A13	0.2	27	.	P1A3
P1	A14	0.3	92	.	P1A4
P1	A15	0.4	147	.	P1A5
P1	A16	0.5	57	.	P1A6
P1	A17	0.6	130	.	P1A7
P1	A18	0.7	6	.	P1A8
P1	A19	0.8	147	.	P1A9
P1	A10	0.9	104	.	P1A0
P1	DAC	100	5000	.	P1DAC
S0	A11	0	84	.	S0A1
S0	A12	0.1	277	.	S0A2
S0	A13	0.2	123	.	S0A3
S0	A14	0.3	395	.	S0A4
S0	A15	0.4	502	.	S0A5
S0	A16	0.5	209	.	S0A6
S0	A17	0.6	1216	.	S0A7
S0	A18	0.7	132	.	S0A8
S0	A19	0.8	502	.	S0A9
S0	A10	0.9	181	.	S0A0
S0	DAC	100	5000	.	S0DAC
S1	A11	0	84	.	S1A1
S1	A12	0.1	277	.	S1A2
S1	A13	0.2	123	.	S1A3
S1	A14	0.3	395	.	S1A4
S1	A15	0.4	502	.	S1A5
S1	A16	0.5	209	.	S1A6
S1	A17	0.6	1216	.	S1A7

S1	A18	0.7	132	.	S1A8
S1	A19	0.8	502	.	S1A9
S1	A10	0.9	181	.	S1A0
S1	DAC	100	5000	.	S1DAC
O0	A11	0	33	.	O0A1
O0	A12	0.1	62	.	O0A2
O0	A13	0.2	36	.	O0A3
O0	A14	0.3	192	.	O0A4
O0	A15	0.4	174	.	O0A5
O0	A16	0.5	64	.	O0A6
O0	A17	0.6	276	.	O0A7
O0	A18	0.7	66	.	O0A8
O0	A19	0.8	174	.	O0A9
O0	A10	0.9	103	.	O0A0
O0	DAC	100	5000	.	O0DAC
O1	A11	0	32	.	O1A1
O1	A12	0.1	62	.	O1A2
O1	A13	0.2	36	.	O1A3
O1	A14	0.3	192	.	O1A4
O1	A15	0.4	174	.	O1A5
O1	A16	0.5	64	.	O1A6
O1	A17	0.6	276	.	O1A7
O1	A18	0.7	66	.	O1A8
O1	A19	0.8	174	.	O1A9
O1	A10	0.9	103	.	O1A0
O1	DAC	100	5000	.	O1DAC
B0	A11	0.0	3	.	B0A1
B0	A12	0.1	12	.	B0A2
B0	A13	0.2	5	.	B0A3
B0	A14	0.3	1	.	B0A4
B0	A15	0.4	0	.	B0A5
B0	A16	0.5	1	.	B0A6
B0	A17	0.6	0	.	B0A7
B0	A18	0.7	5	.	B0A8
B0	A19	0.8	0	.	B0A9
B0	A10	0.9	1	.	B0A0
B0	DAC	100	5000	.	B0DAC
B1	A11	0.0	3	.	B1A1
B1	A12	0.1	12	.	B1A2
B1	A13	0.2	5	.	B1A3
B1	A14	0.3	1	.	B1A4
B1	A15	0.4	0	.	B1A5
B1	A16	0.5	1	.	B1A6
B1	A17	0.6	0	.	B1A7
B1	A18	0.7	5	.	B1A8

```
B1   A19 0.8 0    . B1A9
B1   A10 0.9 1    . B1A0
B1   DAC 100 5000 . B1DAC


SP0  A11 0.0 8    . SP0A1
SP0  A12 0.1 3    . SP0A2
SP0  A13 0.2 0    . SP0A3
SP0  A14 0.3 5    . SP0A4
SP0  A15 0.4 52   . SP0A5
SP0  A16 0.5 9    . SP0A6
SP0  A17 0.6 0    . SP0A7
SP0  A18 0.7 4    . SP0A8
SP0  A19 0.8 52   . SP0A9
SP0  A10 0.9 1    . SP0A0
SP0  DAC 100 5000 . SP0DAC

SP1  A11 0.0 8    . SP1A1
SP1  A12 0.1 3    . SP1A2
SP1  A13 0.2 0    . SP1A3
SP1  A14 0.3 5    . SP1A4
SP1  A15 0.4 52   . SP1A5
SP1  A16 0.5 9    . SP1A6
SP1  A17 0.6 0    . SP1A7
SP1  A18 0.7 4    . SP1A8
SP1  A19 0.8 52   . SP1A9
SP1  A10 0.9 1    . SP1A0
SP1  DAC 100 5000 . SP1DAC

A11  H1  0   100  . A1H1
A12  H2  0   100  . A2H2
A13  H3  0   100  . A3H3
A14  H4  0   100  . A4H4
A15  H5  0   100  . A5H5
A16  H6  0   100  . A6H6
A17  H7  0   100  . A7H7
A18  H8  0   100  . A8H8
A19  H9  0   100  . A9H9
A10  H10 0   100  . A10H10
DAC  DHB 0   1000 . DACDHB

H1   T   0   100  . H1T
H2   T   0   100  . H2T
H3   T   0   100  . H3T
H4   T   0   100  . H4T
H5   T   0   100  . H5T
H6   T   0   100  . H6T
H7   T   0   100  . H7T
```

```
H8   T   0   100  . H8T
H9   T   0   100  . H9T
H10  T   0   100  . H10T
DHB  T   0   1000 . DHBT
;


title3 'Side Constraints';
     data cond1;
input G0A5 G0A9 G1A5 G1A9 P0A5 P0A9 P1A5 P1A9 S0A5 S0A9 S1A5
S1A9 O0A5 O0A9 O1A5 O1A9 B0A5 B0A9 B1A5 B1A9 SP0A5 SP0A9
SP1A5 SP1A9 _type_ $ _rhs_;
cards;
1 1 1 1 . . . . . . . . . . . . . . . . . . . . LE 683
. . . . 1 1 1 1 . . . . . . . . . . . . . . . . LE 147
. . . . . . . . 1 1 1 1 . . . . . . . . . . . . LE 502
. . . . . . . . . . . . 1 1 1 1 . . . . . . . . LE 174
. . . . . . . . . . . . . . . . 1 1 1 1 . . . . LE 0
. . . . . . . . . . . . . . . . . . . . 1 1 1 1 LE 52
;

proc netflow
   nodedata=noded
   arcdata=arcd
   condata=cond1
   conout=solution;


print problem;
proc print data=solution;
sum _fcost_;
```

Appendix E: Data Set Information

The following airports were used as the population of airports for the aircraft-airport assignment problem to consider in all three data sets:

1. Northhampton
2. Albany
3. Charleston
4. Wichita
5. Albuquerque
6. Little Rock
7. Shreveport
8. Oklahoma City
9. Carswell AFB
10. Knoxville
11. Indianapolis
12. Scott AFB
13. Lexington

The following aircraft were generated at the following times on the present day for the first two data sets. The aircraft for the third data set are identical to the first two, but generation starts at time 1.0 (time 0 on the second day).

Aircraft	Time Generated
1	0.0
2	0.1
3	0.2
4	0.3
5	0.4
6	0.5
7	0.6
8	0.7
9	0.8
10	0.9

The following patients were generated for the numbered data sets.

Data Set 1

Patient Category	Number	Time Generated
General Medical	130	0.0
Psychological	30	0.0
Surgical	440	0.0
Orthopedic	360	0.0
Burn	30	0.0
Spinal	10	0.0

Data Set 2

Patient Category	Number	Time Generated
General Medical	130	0.0
Psychological	30	0.0
Surgical	440	0.0
Orthopedic	360	0.0
Burn	20	0.0
Spinal	20	0.0

Data Set 3

Patient Category	Number	Time Generated
General Medical	13	0.0
Psychological	3	0.0
Surgical	44	0.0
Orthopedic	36	0.0
Burn	3	0.0
Spinal	1	0.0
General Medical	130	1.0
Psychological	30	1.0
Surgical	440	1.0
Orthopedic	360	1.0
Burn	30	1.0
Spinal	10	1.0

Appendix F Data Set 1: Iteration 1

GAMS/ZOOM output
Objective function value: 19

Aircraft	Airport Assignment
1	Carswell
2	Northampton
3	Charleston
4	Albuquerque
5	Wichita
6	Oklahoma City
7	Knoxville
8	Lexington
9	Albany
10	Lexington

SAS Output: New Aircraft Loads (A_{jk})
Wait Time: 846.4 days

Aircraft	GM	PYS	SUR	ORTH	SPI	BURN
1	59	0	0	36	0	5
2	38	30	0	31	0	1
3	11	0	46	42	0	1
4	0	0	0	90	10	0
5	0	0	100	0	0	0
6	0	0	99	0	0	1
7	0	0	95	0	0	5
8	0	0	100	0	0	0
9	0	0	0	99	0	1
10	22	0	0	62	0	12
Dummy	0	0	0	0	0	4

Iteration 2

GAMS/ZOOM output
Objective function value: 4

Aircraft	Airport Assignment
1	Northampton
2	Albany
3	Wichita
4	Carswell
5	Carswell
6	Shreveport
7	Albuquerque
8	Albany
9	Lexington
10	Charleston

SAS Output: New Aircraft Loads (A_{jk})
Wait Time: 549.1 days

Aircraft	GM	PYG	SUR	ORTH	SPI	BURN
1	24	0	4	66	1	5
2	70	30	0	0	0	0
3	11	0	46	42	0	1
4	0	0	0	91	9	0
5	0	0	100	0	0	0
6	0	0	96	0	0	4
7	0	0	95	0	0	5
8	0	0	99	0	0	1
9	0	0	0	99	0	1
10	25	0	0	62	0	12
Dummy	0	0	0	0	0	1

Iteration 3

GAMS/ZOOM output
Objective function value: 1

Aircraft	Airport Assignment
1	Northampton
2	Lexington
3	Wichita
4	Carswell
5	Northampton
6	Shreveport
7	Albuquerque
8	Knoxville
9	Albany
10	Charleston

SAS Output: New Aircraft Loads (A_{jk})
Wait Time: 450 days

Aircraft	GM	PYS	SUR	ORTH	SPI	BURN
1	17	6	11	66	0	0
2	0	0	72	27	0	1
3	52	5	0	42	0	1
4	0	19	71	0	10	0
5	0	0	95	0	0	5
6	0	0	96	0	0	4
7	0	0	95	0	0	5
8	35	0	0	64	0	1
9	0	0	0	99	0	1
10	26	0	0	62	0	12
Dummy	0	0	0	0	0	0

Appendix G Data Set 2: Iteration 1

GAMS/ZOOM output
Objective function value: 18

Aircraft	Airport Assignment
1	Northampton
2	Carswell
3	Knoxville
4	Northampton
5	Lexington
6	Carswell
7	Charleston
8	Lexington
9	Albuquerque
10	Carswell

SAS Output: New Aircraft Loads (A_{jk})
Wait Time: 450 days.

Aircraft	GM	PYS	SUR	ORTH	SPI	BURN
1	22	0	12	66	0	0
2	43	30	0	16	11	0
3	2	0	46	42	9	1
4	0	0	95	0	0	5
5	0	0	99	0	0	1
6	0	0	100	0	0	0
7	0	0	88	0	0	12
8	0	0	0	100	0	0
9	63	0	0	36	0	1
10	0	0	0	100	0	0
Dummy	0	0	0	0	0	0

Appendix H Data Set 3: Iteration 1

GAMS/ZOOM output
Objective function value: 27

Aircraft	Airport Assignment
1	Indianapolis
2	Charleston
3	Albuquerque
4	Lexington
5	Carswell
6	Knoxville
7	Scott
8	Northhampton
9	Carswell
10	Albany

SAS Output: New Aircraft Loads (A_{jk})
Wait Time: 10,463 days.

Aircraft	GM	PYS	SUR	ORTH	SPI	BURN
1	79	0	0	16	0	5
2	0	33	66	0	0	1
3	51	0	44	0	0	5
4	0	0	100	0	0	0
5	0	0	63	36	1	0
6	13	0	11	64	10	2
7	0	0	100	0	0	0
8	0	0	100	0	0	0
9	0	0	0	100	0	0
10	0	0	0	99	0	1
Dummy	0	0	0	81	0	19

Bibliography

1. Alfano, Joseph P. and John C. O'Neill, "Wartime CONUS Casualty Distribution System Using Dedicated CRAF Aircraft", MS Thesis/AFIT/GST/OS/85M, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, March 1985 (AD-A156076).

2. Baker, Kenneth R., *Introduction to Sequencing and Scheduling*. New York: John Wiley and Sons Inc, 1974.

3. Balas, Egon and C.H. Martin. "Pivot and Complement - A Heuristic for 0-1 Programming." Management Science. 26: 86-96 (1980).

4. Bazarra, Mokhtar S., John J. Jarvis, and Hanif D. Sherali. *Linear Programming and Network Flows*. New York: John Wiley and Sons Inc, 1990.

5. Becker, John D. "Aeromedical Evacuation -- Do the Pieces Fit?" Air War College Research Report. January 1988 (AD-A202573).

6. Bodin, Lawrence, Bruce Golden, Arjang Assad, and Michael Ball. "Routing and Scheduling of Vehicles and Crews: The State of the Art," Computers and Operations Research. 10: 65-211 (1983).

7. Brooke, Anthony, David Kendrick, and Alexander Meeraus. *GAMS A Users Guide*. Redwood City, CA: The Scientific Press, 1988.

8. Burnes, Michael D. "Application of Vehicle Routing Heuristic to an Aeromedical Airlift Problem." MS thesis, AFIT/GST/ENS/90M-3. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, March 1990 (AD-A220943).

9. Carter, William B. "Allocation and Routing of CRAF MD80 Aircraft." MS thesis, AFIT/GST/ENS/90M-4. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, March 1990 (AD-A220182).

10. Durham, Randy P. "Red Cross Rising." Airlift Summer: 5-9 (Summer 1988).

11. Litko, Joseph. Chief, Command Analysis Group, Air Mobility Command. Scott AFB, IL. Personal Interview. 21 September 1992.

12. Mahlem, Philip. Surgeon General Office, Air Mobility Command. Scott AFB, IL. Personal Interview. 21 September 1992.

13. Nemhauser, George L. and Laurence A. Woolsey. *Integer and Combinatorial Optimization*. New York: John Wiley and Sons, 1988.

14. Parker, R. Gary and Ronald L. Rardin. "An Overview of Complexity Theory in Discrete Optimization." Parts 1 and 2. IIE Transactions. (March, June 1982).

15. Pilkington, Gilbert J. Assistant Chief Medical Plans, Scott AFB, IL. Personal Correspondence. 31 May 1989.

16. Ravindran, A., Don T. Phillips, and James J. Solberg. *Operations Research Principles and Practice*. New York: John Wiley and Sons, 1987.

17. Silver, Edward A. and Rein Peterson. *Decision Systems for Inventory Management and Production Planning*. New York: John Wiley and Sons, 1979.

18. Ware, Keith. Command Analysis Group, Scott AFB, IL. Telephone Interview. 6 July 1992.

19. Ware, Kieth. MD-80 Requirements Analysis. Command Briefing. Scott AFB,IL. 1990.

20. Winston, Wayne L. *Introduction to Mathematical Programming Applications and Algorithms*. Boston: PWS-Kent Publishing Company, 1991.

21. Wolfe, Charles R. "The Use of Simulation to Evaluate Strategic Aeromedical Evacuation Policy and Planning," MS Thesis/AFIT/GOR/ENS/93M-26, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, March 1993.

22. 1st Aeromedical Evacuation Squadron. Contingency Aeromedical Evacuation: A Primer. Pope AFB, NC. 1 May 1992.

Vita

Major Michael J. Loftus was born on 20 June 1958 in Marianna, Florida. He graduated from high school in Austin, Texas in 1976. He attended the United States Air Force Academy, from which he received the degree of Bachelor of Science in Operations Research. Upon graduation he was commissioned in the United States Air Force. He completed pilot training and received his wings at Vance AFB, Oklahoma in August 1981. He then served as an O-2A pilot at Howard AFB, Republic of Panama and as a T-33A instructor pilot and flight commander at Tyndall AFB, Florida. He was then assigned as an instructor pilot and assistant chief pilot at the 50th Tactical Airlift Squadron, Little Rock AFB, Arkansas. He deployed with the squadron to the United Arab Emirates for seven months during Operation Desert Shield and Desert Storm and returned in March of 1991. He then entered the Air Force Institute of Technology in August of 1991.

Permanent address: 1706 Freedom Lane
Copperas Cove, Texas 76522

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (*Leave blank*)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	March 1993	Master's Thesis

4. TITLE AND SUBTITLE

PATIENT SCHEDULING AND AIRCRAFT ROUTING FOR STRATEGIC AEROMEDICAL EVACUATION

5. FUNDING NUMBERS

6. AUTHOR(S)

Michael J. Loftus, Major, US Air Force

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Institute of Technology, WPAFB, OH 45433-6583

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT/GST/ENS/93M-07

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

AMC/XPYR
Scott AFB, IL

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT (*Maximum 200 words*)

The objective of this study is to develop an approach that would assign patients to aeromedical evacuation aircraft and route each aircraft to a single CONUS airport. The goal is to minimize total patient wait time.

A heuristic algorithm is developed which exploits the structure of the problem. The first subproblem solved is the assignment of aircraft to airports. This subproblem is solved using integer programming and the results are input into the second subproblem. The second subproblem is the patient to aircraft assignment problem. This subproblem minimizes patient wait time and is solved using network flow programming. The two subproblems are linked into one algorithm and solved iteratively until termination criteria are met.

The algorithm is tested on three data sets. The results indicate that the algorithm is an efficient method for scheduling patients and routing aircraft, although optimality is not guaranteed.

14. SUBJECT TERMS	15. NUMBER OF PAGES
Aeromedical Evacuation, Integer Program, Minimum Cost Flow, Heuristic Algorithm	100
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

END

FILMED

DATE: 4-93

DTIC